PLC based Robot Manipulator Control using Position based and Image based Algorithm

Patil, Madhav[1]

[1] University Of Bridgeport

Received: 6 December 2016 Accepted: 5 January 2017 Published: 15 January 2017

Abstract

Programmable Logic Controller (PLC) is making big role in the field of automation and robotics. This paper described the design and implementation of Programmable Logic Controller (PLC) based robot manipulator control using two different artificial intelligence algorithms - Position Based and Imaged Based algorithm. The controlled robot is 5 degrees of freedom (DOF) manipulator with a closed kinematic chain, designed for high-performance pick and place applications. The control software is fully developed on a commercial PLC system, using its standard programming tools and the multi-tasking features of its operating system. In particular, this paper analyze in detail the drawbacks and the advantages related to the choice of standard PLCs in this kind of applications, compared to the much common choice of specialized hardware or industrial personal computers, with particular emphasis on the computational performances obtained with the proposed control architecture.

Index terms— artificial intelligence, vision system, programmable logic controller, robot manipulator, industrial automation, design, hardware.

1 INTRODUCTION

obot manipulators are largely used in packaging industry, especially for pick and place operations and box filling. In particular, packaging of small food products, like cookies or candies, requires high performance robots with short cycle times and precise motion control, even if their workspace is relatively small [1]. Such performances can be obtained by means of lightweight parallel-driven or delta-like kinematics, whose advantages in terms of reduced moving masses and inertias are well-known. These robots include a vision system to identify and localize large and unordered products on the conveyors. Typically robot controller is proprietary and limits its access for customers to extend its usage without support from the manufacturer. Since the integration of a robotic system with additional application-specific tools and features often requires the development of software or hardware that closely matches with the basic robot motion control system. In general, customers search for openness of robot controllers for addition or modification of its functionalities. In literature, several open architectures for robot controls has been developed by academic research that limits its application to a smaller scale [2], [3], [4]. In most cases, these architectures take Computers (PC). On one hand, the use of PCs for robot control reduces software development costs that use high level languages and well-designed Integrated Development Environments (IDE). On the other hand, standard PCs doesn't meet the reliability required by complex industrial environments. In summary, the use of more protective enclosures and usage of robust electronic components increase the cost of PC based robot controllers.

Programmable Logic Controllers (PLCs) is a typically a control device playing a dominant role in industrial automation. PLCs provides higher degree of robustness, cheaper than alternative options and ease of use [1]. Electronic components in PLCs are benchmarked for their reliability and robustness ultimately guaranteeing high performance. For this reasons, PLCs are commonly considered as low-level systems, whose main purpose is to control using simple Boolean signals (i.e. discrete control and sequencing) and to supervise the safety and integrity of plants and operators. However, modern PLCs have sufficient computational power to perform complex mathematical calculations using various programming languages (i.e. IEC 61131-3 [5]). This allows developers to implement various algorithms for robot control on PLCs.

Typically, most PLCs support only their proprietary IDE. The respective IDE provides an option for users to custom build/modify their applications which is a limiting factor in PC-based approaches [3]. Therefore, PLC programs can be updated and extended by end-users even without the original source code, which makes PLCs as "open" systems. Despite of these features, PLC-based control systems for robot manipulators are quite rare and, in general, limited to simple Cartesian or gantry-like structures [7].

A robot arm is the combination of links and joints in the form of a chain with one end is fixed while the other end can be moved with certain degree of freedom in native axis of the arm and is termed as end effector. The joints are either prismatic or revolute, driven by actuators. In order to move the end effector along a certain path, the respective joints should be moved appropriately [8]. In this process, it is necessary to perform inverse kinematics equation. In case of redundant manipulator, inverse kinematics is more challenging when compared to a non-redundant manipulator whose kinematics is not so complicated [7].

There are traditional methods such as algebraic solutions, geometric solutions and iterative solutions in order to solve the inverse kinematics problem [5]. However, these methods are computationally complex and exhibits higher execution times. Recently, the particle swarm optimization (PSO) has been successfully applied to various optimization problems. This new optimization algorithm combines social psychology principles in socio-cognition human agents and evolutionary computations [9].

In this paper, authors discusses about the capabilities of a commercial PLCs and its multi-tasking operating system to implement a robot control system including inverse kinematics for on-line trajectory planning. Our design is an integration of vision systems with Robot manipulator and PLC used for implementing Imaged Based Intelligence algorithm. This system design provides location information of object using Position Based algorithm. The particle swarm optimization is used for Position based algorithm as it is characterized as a simple concept. This methodology is easy to implement and computationally efficient [9,10]. The robotic platform described in the paper is designed and developed in Programmable Logic Controller laboratory at University of Bridgeport, CT, USA [1]. The rest of the paper is organized as follows: Sec. II describes the problem definition; Sec. III provides details about the solution, add Sec. IV describes the hardware setup. Sec. V describes about the result and Sec VI describes the conclusion and the future works.

2 II.

3 PROBLEM DEFINITION

Increase in automation needs with revolutionary advancement in technology motivates the researchers to develop next generation automation applications [11]. Typically industrial processes uses manipulator arms for picking and placing the objects in close proximity. This involves robot arms to repeatedly perform movements with high accuracy and with precise joint angles [12]. The common industrial manipulator is often referred as robot arm with joints and angles as shown in Figure 1. The robot arm used in our application is an articulated arm consists of all revolute joints. The articulated robot arm has maximum flexibility and can reach over and under the objects. As all joints are revolute these robots can cover large workspace and are easy to seal. The robot manipulators are assigned to accomplish the specific task in unstructured environment with minimal joint movements and with best shortest path.

4 a) State Space

As the manipulator is designed for pick and place application, the manipulator picks up the object and places it in relative positions such and are termed as good position or bad position or rejection area according to the sensory data. So finding the states for the robot manipulator is finite. The arm has its work space and can reach to those positions by various paths.

5 b) Initial State

The initial state for the robot manipulator could be any state depends on the signal send by the sensors to the arm. But at the start of the operations the manipulator always go to the home or nesting position.

6 c) Action

The manipulator action depend on the perceived signal from the sensors (camera, part detect sensor, etc.). The manipulator takes action based on the sensor data to the controller and controller performs the movement of arm. So manipulator moves toward the destination area by avoiding the obstacles in the path in a given time limit with respective speed. To reach the object, the manipulator has to find out the best possible path with minimal joint angle movement within its work space. The end effector will try to reach the object as soon as position coordinates are calculated. There are also other actions performed such as for open and close the gripper, take snap shot of parts, start/stop the conveyor, start/stop the motor etc. The robot motion path planning has been studied more than two decades [13]. Deriving the best possible inverse kinematic solution for end effector is challenging and difficult. Some of the challenging aspects in designing reverse kinematic solution are: Year 2017

7 H d) Transition Model

The transition model for the manipulator depends on the action taken. In the start, the manipulator returns to the homing position. So we consider the initial state as the homing position, but not all the time. The camera mounted on the end effector will continuously feed the current position coordinates of the robot manipulator to the controller and the algorithm controls the motion of robot arm. The complete transition model is shown in Figure 2.

8 e) Goal test

The goal test of this model is to pick and place the objects/parts with respective area for further processing by detecting metal and nonmetallic parts and avoiding the obstacle or collision of the robot arm with any other part of the system.

9 f) Path Cost

The path cost of this application varies according to the product cost, i.e. parts used for the system assembling, and other factors.

The complete flow chart of the application designed and implemented as given in Figure 3.

10 H

To achieve the goal in this study, the position trajectory calculation of the robot manipulator is the most challenging task, as it has to avoid the obstacle and reach the product for picking [14]. As stated in above problem transition state for the robot manipulator has various possible paths to reach object. Among those paths, the best path will have the minimal joint movements and shortest distance within its workspace. Many techniques have been developed for finding the inverse kinematics of the manipulator. The complexity of finding the kinematics solution increases with number of joints or Degree Of Freedom (DOF). The robot manipulator position, path planning and motion control in 3 dimensional workspace become a key factor for control system design engineers and robot manufacturers. To achieve this functionality, the robot arm should be self-proficient, flexible, low power consumptions, fully efficient. One of the challenging task for robot arm in industry is to move its end effector from initial position to desired position in working environment with least residual vibration, minimal torque, obstacle avoidance and collision free kinematics, shortest time interval and/or distance in a desired path [15].

Dealing with complex, higher level control system with continuous interactive subsystem in dynamic environment is difficult and requires sophisticated and intellectual controllers with continuous process optimization. In this study, we proposed the solution for path planning as stated in the previous section. The proposed solution has been applied and tested on robot manipulator with 5 DOF. The experimental set up is explained in Section IV. In this experiment we are using position based algorithm in combination with Image based algorithm to finding the best possible solution for path planning. In this study, we are implementing Artificial Intelligence in the robot arm control system using PLC as shown in Figure 4. The system is completely knowledge based as it uses information from the sensors and performs the action using robot arm and actuators. The position based search algorithm uses the visual data provided by the imaged based algorithm and calculates the joint velocity and angles to form the inverse kinematic solution in 3 dimensional workspace [16]. By adding vision or imaged based algorithm, the robotic control system is more flexible, adaptable and increases the accuracy in the joints movement.

The principal advantage of using positionbased control is the chance of defining tasks in a standard Cartesian frame. On the contrary, the imagebased control is less sensitive to calibration errors; however, it requires online calculation of the image Jacobean that is a quantity depending on the distance between the target and the camera which is difficult to evaluate. A control of a manipulator in the image plane when mapped with the joint space is strongly nonlinear and may cause problems when crossing those points which are singular for the kinematics of the manipulator. The main purpose or goal of this study is to use the visual feedback from images captured by the camera and the Cartesian space co-ordinates of the target object which ultimately controls the motion of the robot to perform a task. The starting position and coordinate frame boundary is taught to the robot arm. From the calculated co-ordinates of the target object, the surface model in 3 dimensional co-ordinate systems will be constructed for robot manipulator. The sensor data will be used for knowledge base and will be used to avoid collision as well as can be used to find optimal shortest possible path or trajectory by checking each point in its workspace. The sensors can give a signal when contact is made with obstacles, detect metal or nonmetal objects, or measure a force being applied. Due to this knowledge provided by sensory units to the system, robot path can be planned before its execution to the target position.

Firstly, a camera is mounted on a manipulator end effector and it catches a 2-D image, a true potential can be exactly calculated. We assign the two dimensional coordinate system with the x-axis and yaxis forming a basis for the image plane and the z-axis perpendicular to the image plane. The origin located at distance ? behind the image plane as shown in Figure 5. In order to determine the position of the target object in the image plane, camera will capture the images and through which only the central point (only a single pixel) have to be identified, without knowing this point's co-ordinates in the attached Cartesian reference system. This coordinates will be feed to the Position Based Algorithm for calculating the joints angle and trajectory. The position finding algorithm outputs the xcoordinate and y-coordinate in the image frame along with a scaling factor (?). The scaling factor is related to the dimensions of the gripper and is used to get an idea of the elevation of the gripper. This helps in making the gripper co-planar with the target object.

We implemented our proposed algorithms and optimal path planning schemes on a PLC based Robot arm control system for sorting, pick and place application as shown in Figure 6. The Mitsubishi RV-M1 robot is driven with robot drive unit Movemaster EX from which it interfaces with Mitsubishi PLC with help of 50 pin connector and to an another interface for robot teach pendant. This drive unit stores the code in the robot. The nomenclature of the robot arm is shown in Figure 7. The application we used for testing these two algorithm with PLC is Pick and Place application with Quality Inspection. As an initial step we localize the object using a traditional Haar classifier. We trained multiple models for recognition various objects in the scene. At time of training process each Haar object detection model will be fed with respective positive samples or in class samples and out of class samples. We then evaluate the performance of the trained model using test data. The processing is performed on a windows machine using Matlab Software. We chose this method (Haar classifier) as it is one of the successful and simple object detection method used widely in machine vision technologies. Some of the sample training images are shown in below Figure 9, Figure 10, Figure 11, and Figure 12.

The pseudo code for Experimental Setup is as given bellow. products with unique features to distinguish whether it is good or bad product.

11 JOB 1

We used relay for job 1, it will check whether screw in the socket is present or not. If cognex checker detects screw on the right position then it will consider it as good part and send it to accept position else it will send it to reject position. We used holder for job 2, checker will check whether holding pins are assembled in holder during manufacturing process or not. If all holding pins are present in holder then checker will consider it as good product else it will consider it as bad product and send the good product to accept position and bad product to reject position. Purell hand sanitizer is used in job 3. In this job we check the label of Purell brand name in the hand sanitizer container. Container with Purell brand name is considered as a good product and without Purell brand name is considered as a bad product. We used nut for job 4. Nut with two marks is considered as good product and nut without marks will be considered as a bad part. [1] [2]

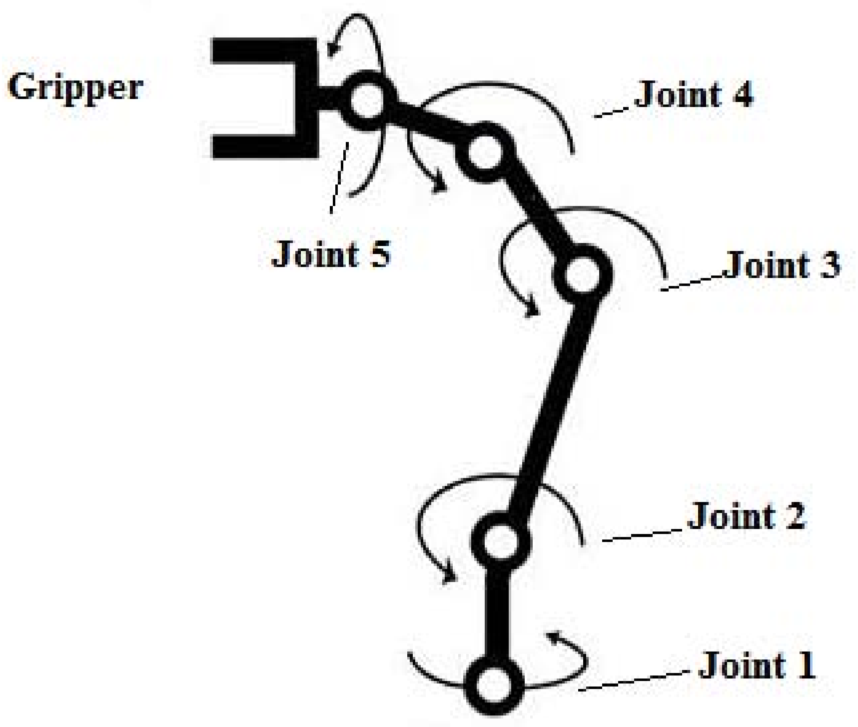
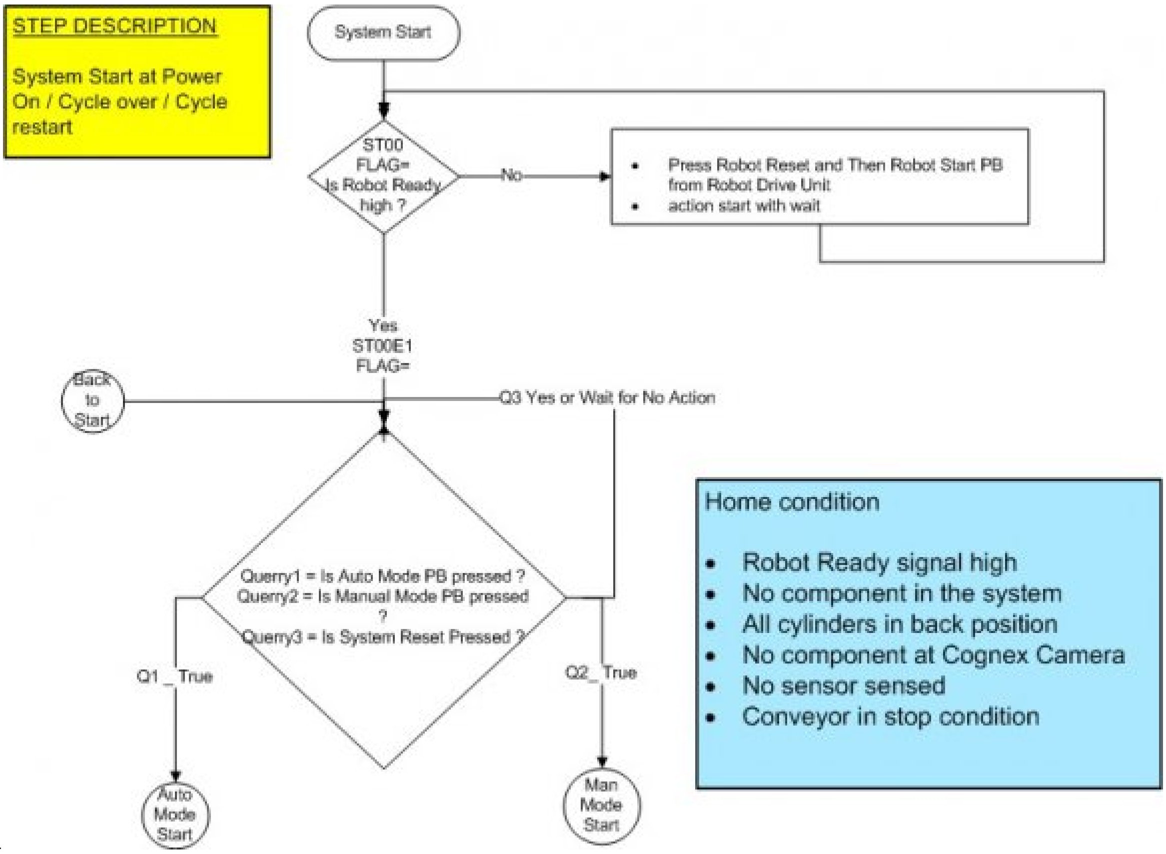


Figure 1:



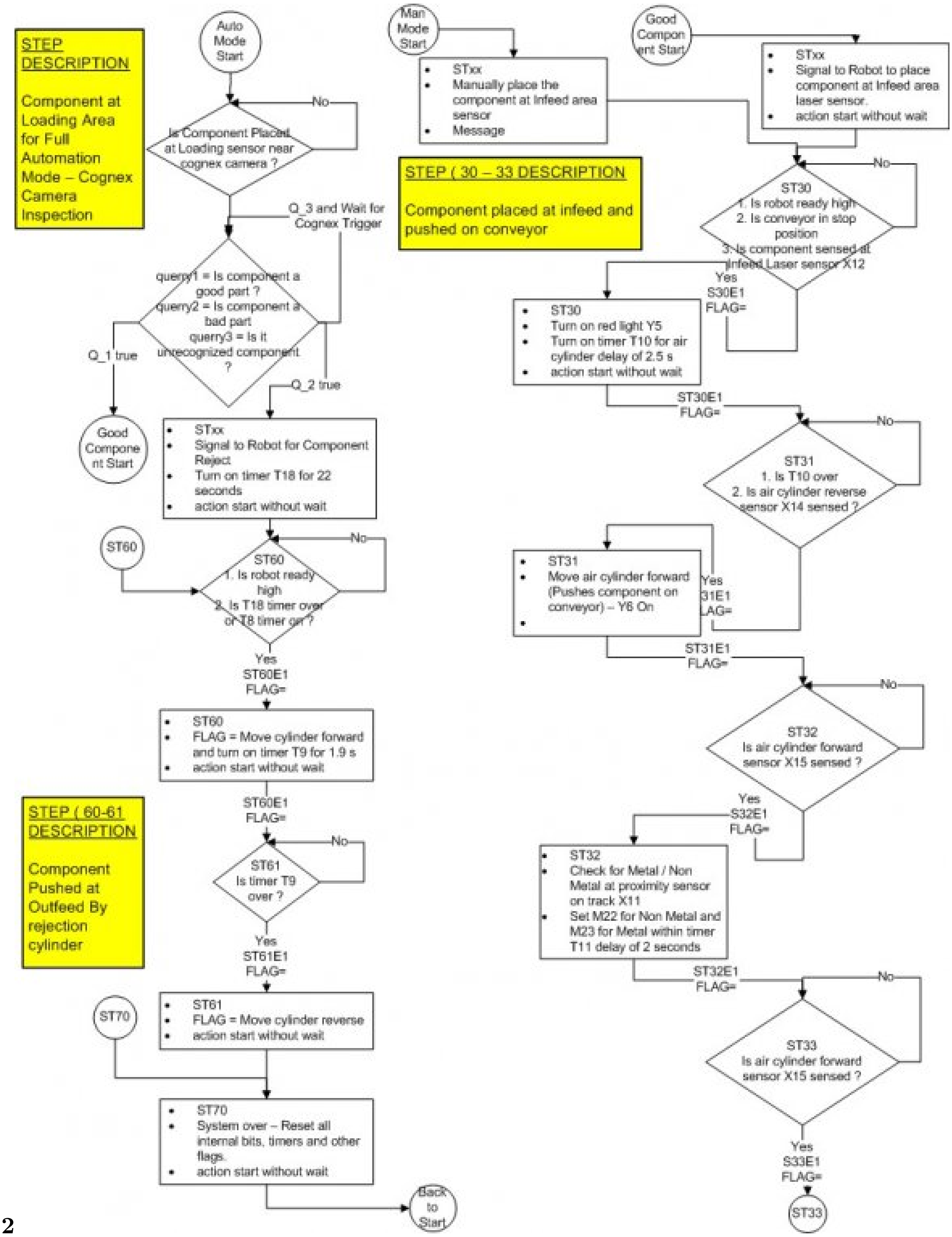
1

Figure 2: Figure 1 :

188

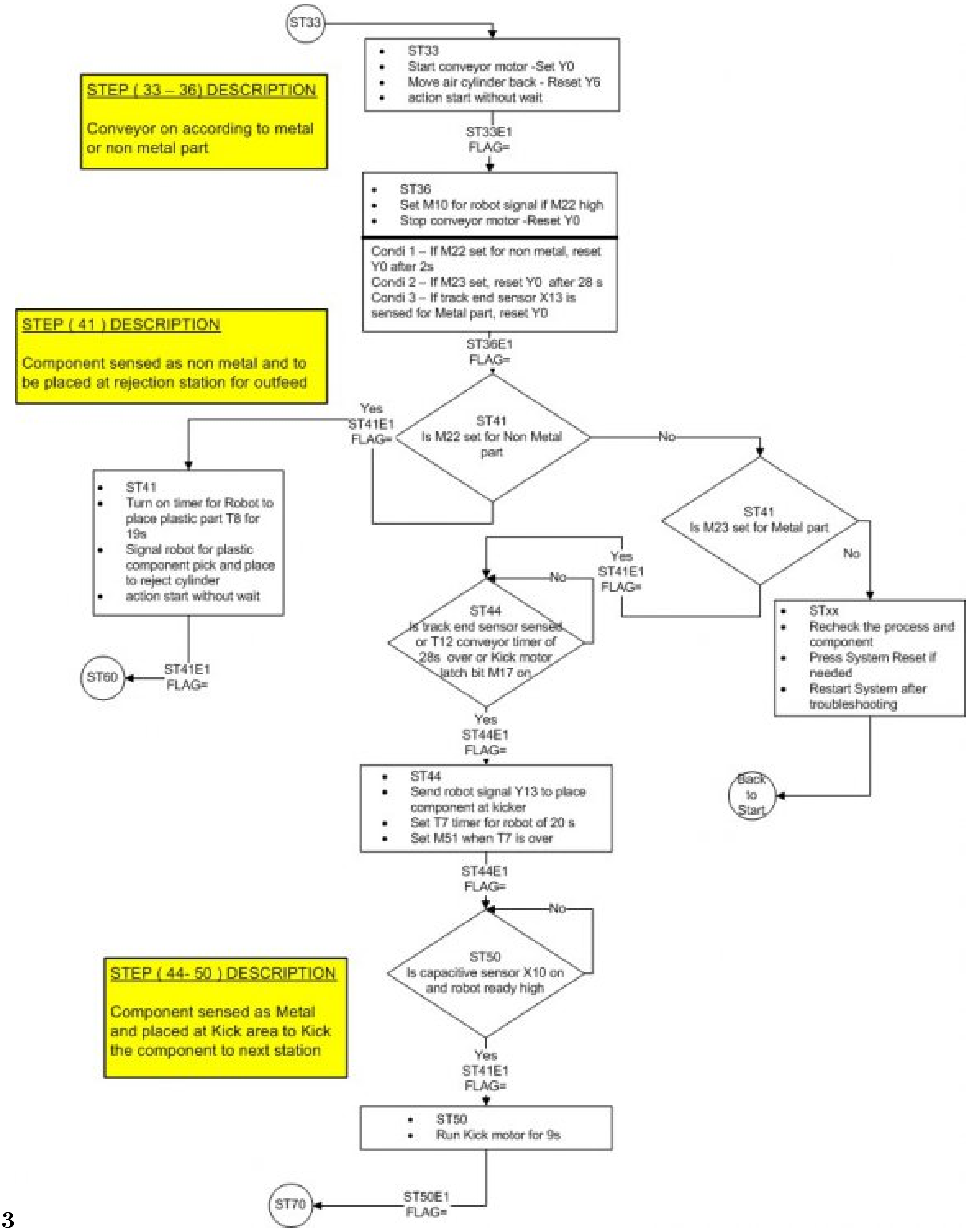
[1]© 2017 Global Journals Inc. (US)

[2]PLC based Robot Manipulator Control using Position Based and Image Based Algorithm © 2017 Global Journals Inc. (US)



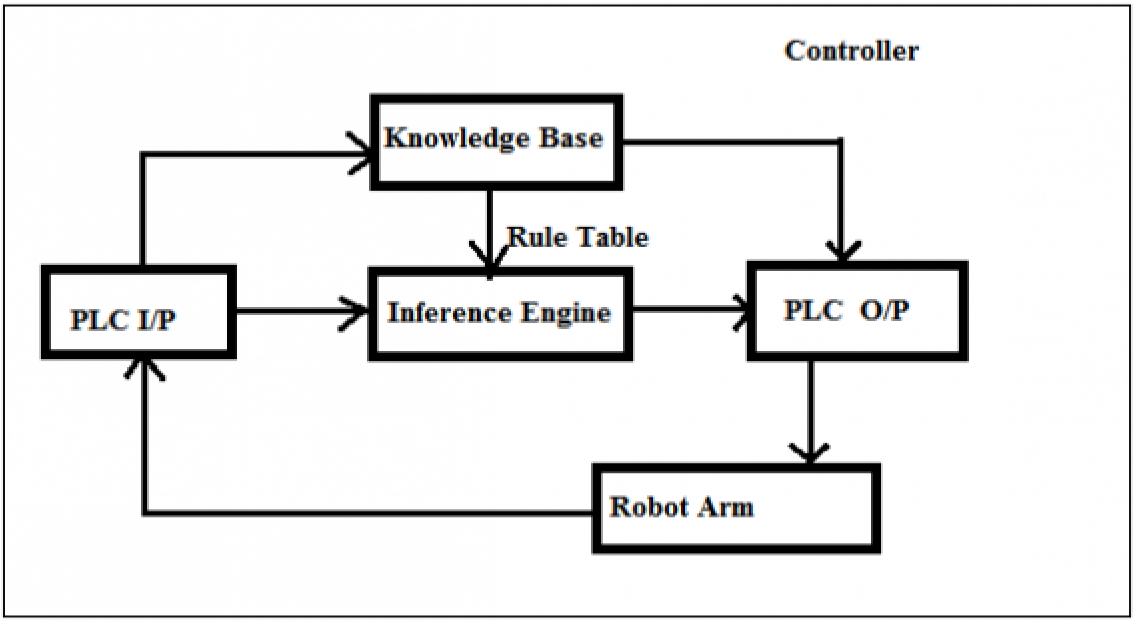
2

Figure 3: Figure 2 :



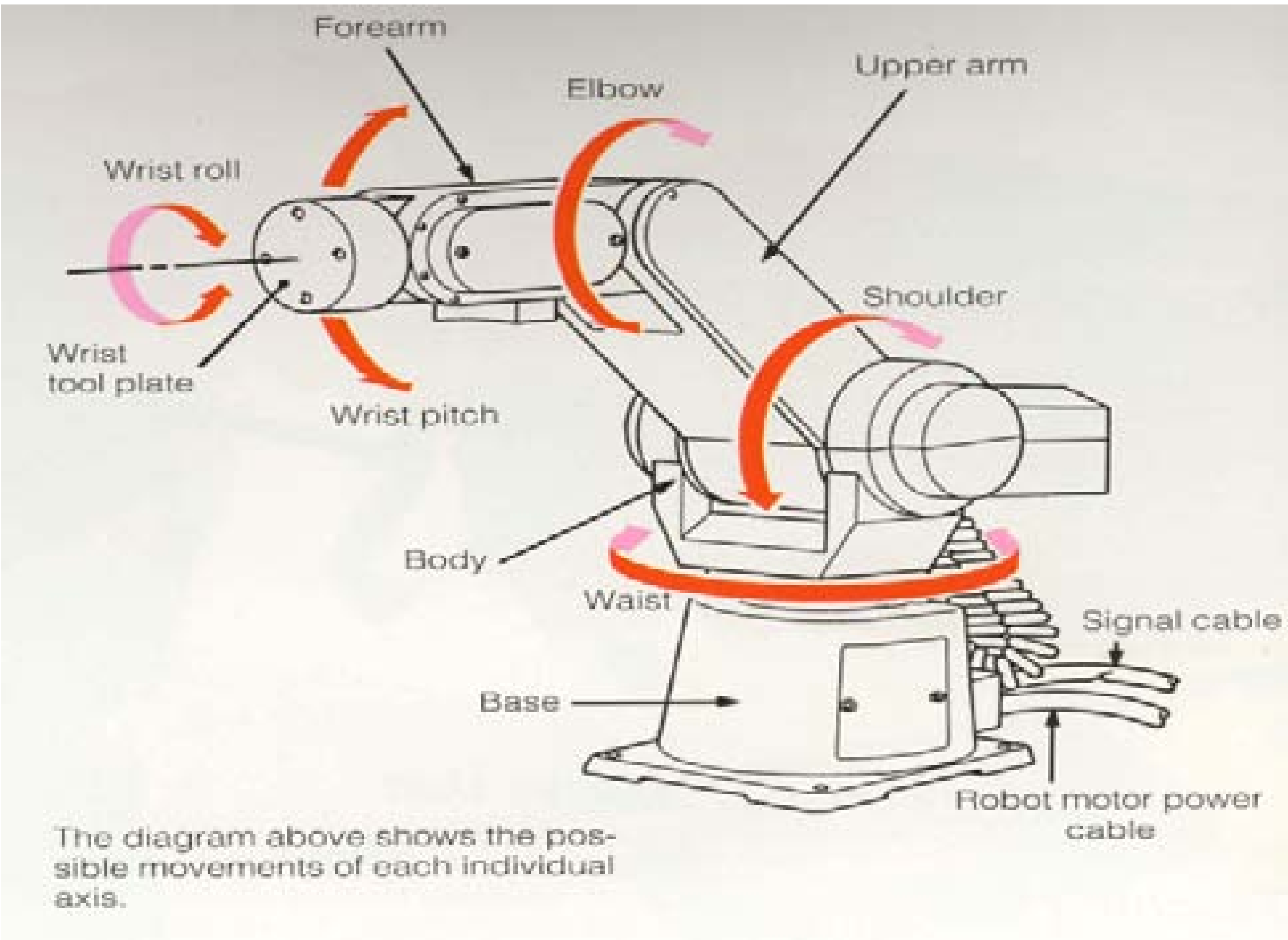
3

Figure 4: Figure 3 :



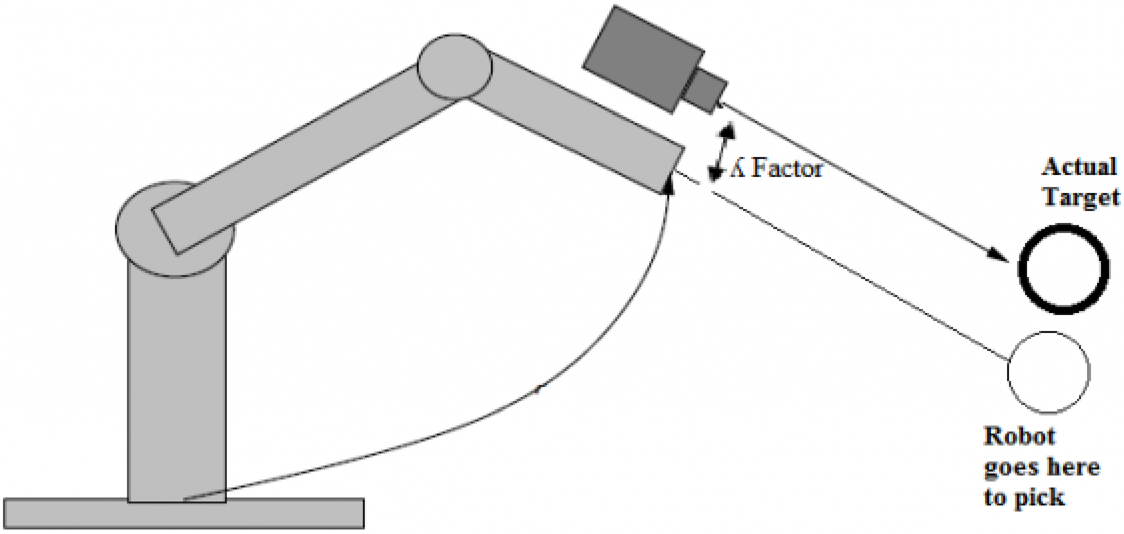
4

Figure 5: Figure 4 :



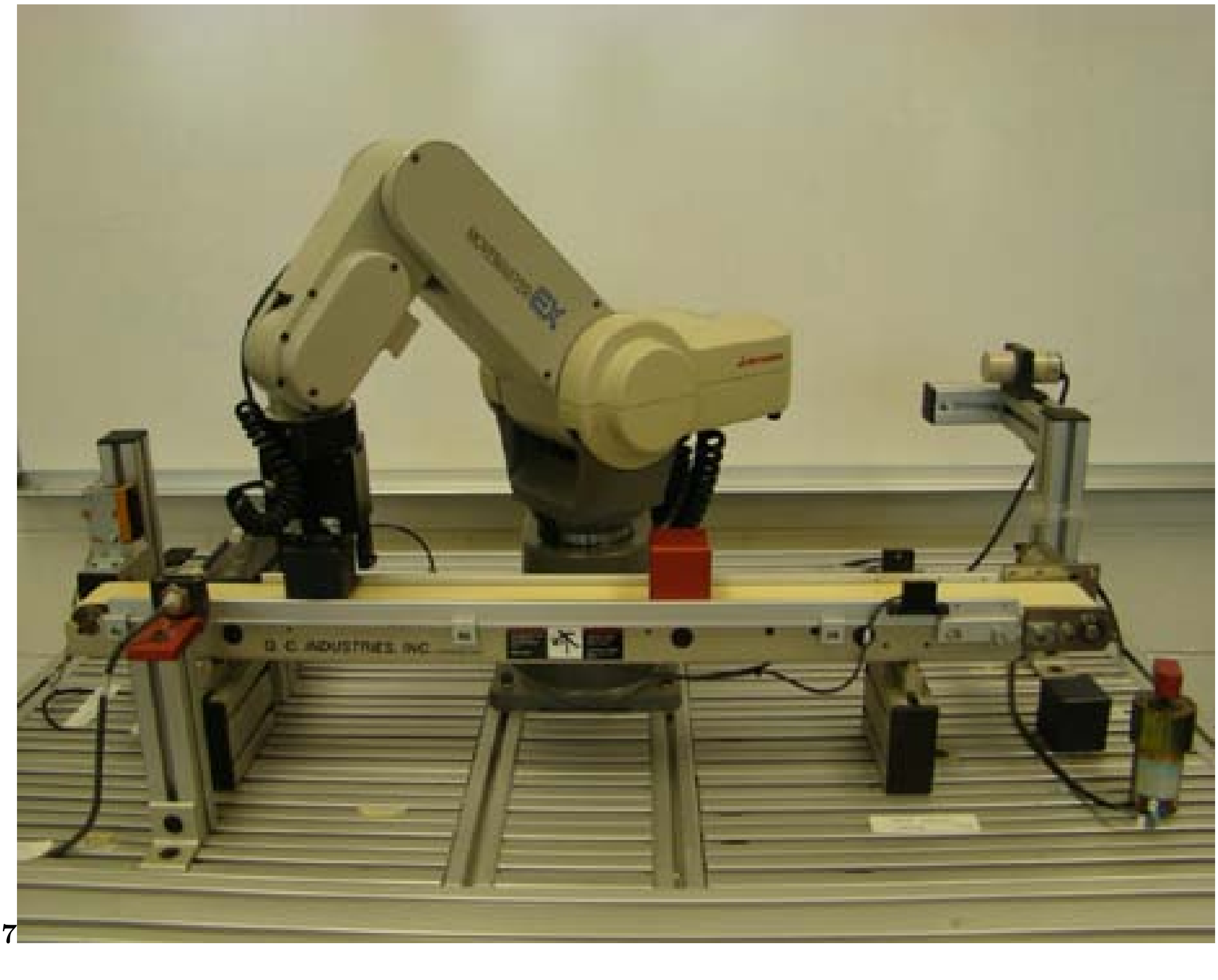
5

Figure 6: Figure 5 :



6

Figure 7: Figure 6 :



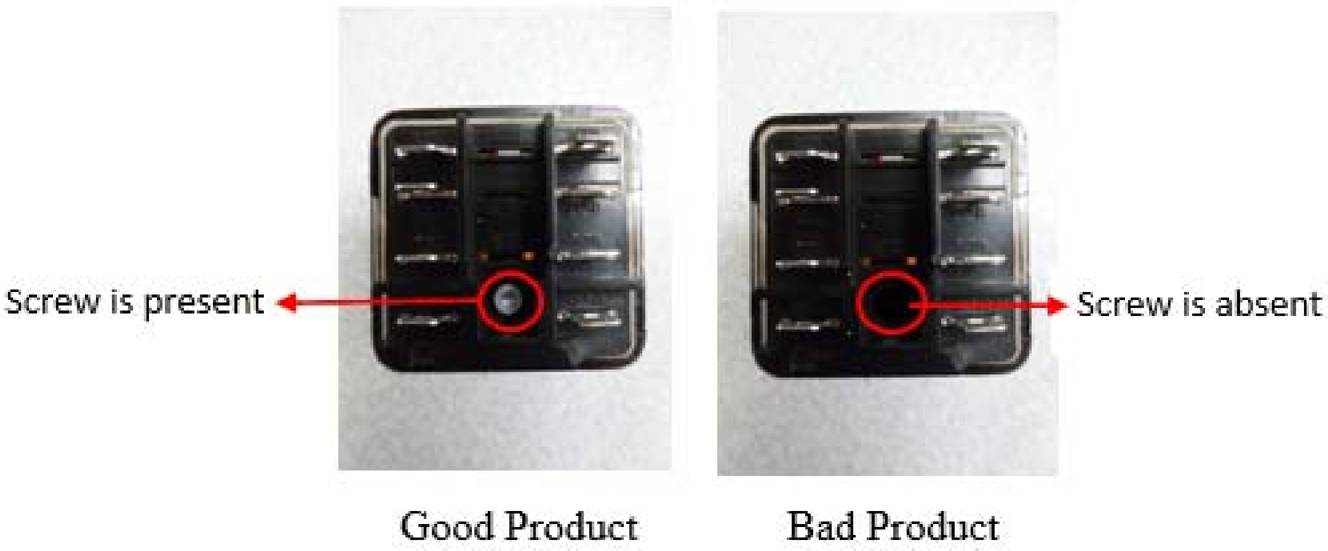
7

Figure 8: Figure 7 :

MOVEMASTER RV-M1		
Construction		Vertical, articulated
Degrees of freedom		5 (not including hand)
Electrical drive system		DC servo motors
Reach		250 + 160 mm
Operation range	Waist rotation	300° (max. 120°/s)
	Shoulder rotation	130° (max. 72°/s)
	Elbow rotation	110° (max. 109°/s)
	Wrist pitch	± 90° (max. 100°/s)
	Wrist roll	± 180° (max. 163°/s)
Maximum path velocity		1,000 mm/s (PTP at wrist tool plate)
Lifting capacity		1.2 kg incl. hand
Position repeatability		± 0.3 mm at wrist tool plate
Position direction		Limit switches and encoders
Installation position		Horizontal
Ambient temperature		5° C - 40° C
Weight		19 kg

8

Figure 9: Figure 8 :



9

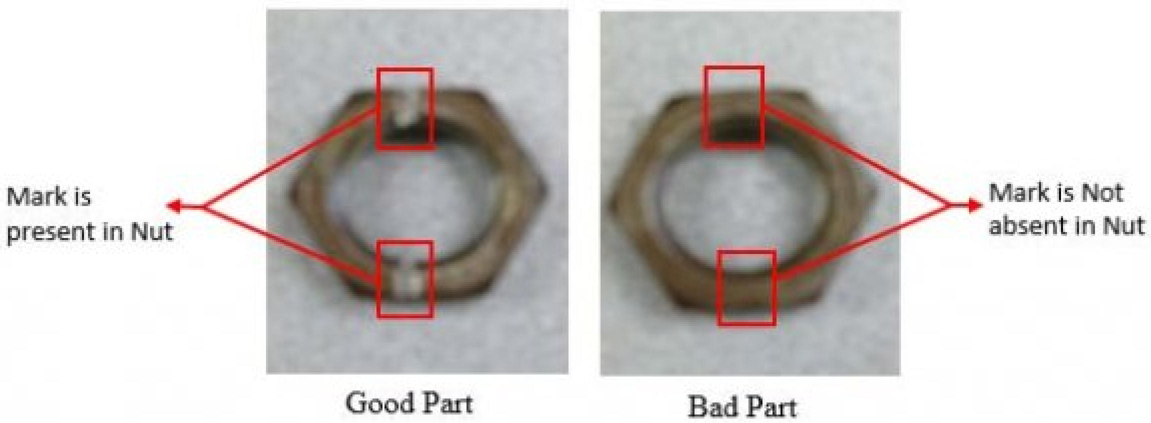
Figure 10: Figure 9 :

10

Figure 11: Figure 10 :

11

Figure 12: Figure 11 :



12

Figure 13: Figure 12 :

	2. begin (At start node Get the unvisited adjacent)
	while (there are bNODEs proceed with PBA and IBA algorithm and generate Step code for the branch till you reach a tNODE or xNODE)
	if (xNODE proceed to 3)
	else if (tNODE go to 7)

V. RESULTS	else go to 10 3. Global Journal 31 Year 2017 of Researches in Engineering ( ) Volume XVII Issue I Version I H
1. Initialize: Start node Stack = empty End node Stack = empty All other variables initialized.	

Figure 14:

.1 Global

.2 CONCLUSION

In this paper we proposed an algorithms and implementation method for calculating the inverse kinematic solution and trajectory planning for industrial robot manipulator using the Position based algorithm in combination with Image based algorithm. The implementation is carried out on Movemaster Robot Arm with PLC for sorting metal and nonmetal objects, pick and place application. The proposed combination algorithm reduces the computation time and positioning error for finding the target in real time.

[Palmieri et al. ()] 'A Comparison between Position-Based and Image-Based Dynamic Visual Servoings in the Control of a Translating Parallel Manipulator'. G Palmieri , M Palpacelli , M Battistelli , M Callegari . *Journal of Robotics* 2012. 2012 p. 11.

[Abukhalil et al. ()] 'A Comprehensive Survey on Decentralized Modular Swarm Robotic systems and Deployment Environments'. T Abukhalil , M Patil , T Sobh . *International Journal of Engineering (IJE)* 2013. 7 p. .

[Macchelli and Melchiorri (2002)] 'A real-time control system for industrial robots and control applications based on real-time Linux'. A Macchelli , C Melchiorri . *15th IFAC World Congress*, (Barcelona, Spain) July 2002.

[Arvin et al. ()] 'A short-range infrared communication for swarm mobile robots'. F Arvin , S Khairulmizam , R Abdul Rahman . *2009 International Conference on Signal Processing Systems*, 2009.

[Asada and Youcef-Toumi (1983)] 'Analysis and design of semidirect drive robot arms'. H Asada , K Youcef-Toumi . *Proc. American Control Conference*, (American Control ConferenceSan Francisco, CA) June 1983.

[Abukhalil et al. ()] 'Coordinating a heterogeneous robot swarm using Robot Utility-based Task Assignment (RUTA)'. T Abukhalil , M Patil , S Patel , T Sobh . *2016 IEEE 14th International Workshop on Advanced Motion Control (AMC)*, (Auckland, New Zealand) 2016.

[Christensen et al. ()] 'From fireflies to fault-tolerant swarms of robots'. A L Christensen , R Ogrady , M Dorigo . *IEEE Transactions on Evolutionary Computation* 2009. 13 (4) p. .

[Patil et al. ()] 'Hardware Architecture Review of Swarm Robotics System: Self-Reconfigurability, Self-Reassembly, and Self-Replication'. M Patil , T Abukhalil , T Sobh . *ISRN Robotics* 2013. 2013 p. 11.

[Whitney and Lozinski ()] 'Industrial robot forward calibration method and results'. D Whitney , A Lozinski . *ASME Journal of Dynamic Systems, Measurement and Control* 1986. 108 p. .

[Patil and Toporovsky ()] 'Integration of Vision System, Intelligent ROBO Actuator, HMI and PLC to Design a Universal Quality Inspection or Control Machine," i-manager's'. M Patil , J Toporovsky . *Journal on Mechanical Engineering* 2012. 2 (3) p. .

[Hashimoto et al. ()] 'Manipulator control with image-based visual servo'. K Hashimoto , T Kimoto , T Ebine , H Kimura . *IEEE International Conference on Robotics and Automation*, (Sacramento, CA, USA) 1991.

[Borenstein et al. ()] 'Mobile Robot Positioning -Sensors and Techniques'. J Borenstein , H Everett , L Feng , D Wehe . *Journal of Robotic Systems* 1996. 14 (4) p. .

[Bonfe et al. ()] 'PLC-based control of a robot manipulator with closed kinematic chain'. M Bonfe , M Vignali , M Fiorini . *IEEE International Conference in Robotics and Automation*, 2009.

[Kao and Gong ()] 'Robot-based computerintegrated manufacturing as applied in manufacturing automation'. I Kao , C Gong . *Robotics and Computer-Integrated Manufacturing* 1997. 13 (2) p. .

[Patil et al. ()] 'UB robot swarm -Design, implementation, and power management'. M Patil , T Abukhalil , S Patel , T Sobh . *12th IEEE International Conference on Control and Automation (ICCA)*, (Kathmandu) 2016.

[Patil et al. (2016)] 'UB Swarm: Hardware Implementation of Heterogeneous swarm robot with fault detection and power management'. M Patil , T Abukhalil , S Patel , T Sobh . *International Journal of Computing* September 2016. 15 (3) p. .