International Journal of Electrical, Electronic and Communication Sciences
ISSN: 2517-9438
Vol:2, No:5, 2008

PUMA 560 Optimal Trajectory Control using Genetic Algorithm, Simulated Annealing and Generalized Pattern Search Techniques

Sufian Ashraf Mazhari and Surendra Kumar, Member IEEE

***Abstract*—**Robot manipulators are highly coupled nonlinear systems, therefore real system and mathematical model of dynamics used for control system design are not same. Hence, fine-tuning of controller is always needed. For better tuning fast simulation speed is desired. Since, Matlab incorporates LAPACK to increase the speed and complexity of matrix computation, dynamics, forward and inverse kinematics of PUMA 560 is modeled on Matlab/Simulink in such a way that all operations are matrix based which give very less simulation time. This paper compares PID parameter tuning using Genetic Algorithm, Simulated Annealing, Generalized Pattern Search (GPS) and Hybrid Search techniques. Controller performances for all these methods are compared in terms of joint space ITSE and cartesian space ISE for tracking circular and butterfly trajectories. Disturbance signal is added to check robustness of controller. GA-GPS hybrid search technique is showing best results for tuning PID controller parameters in terms of ITSE and robustness.

***Keywords*—**Controller Tuning, Genetic Algorithm, Pattern Search, Robotic Controller, Simulated Annealing.

I. Introduction

THE term Robot has been applied to wide variety of mechanical devices. An important class of Robots is the manipulator arm such as Puma 560 Robot. These arms are widely used for material handling, welding, assembling, painting, grinding and other industrial applications. These applications may require path planning, trajectory generation and control design [1-3]. Path planning consists of determining a curve in the workspace, connecting the initial and final desired position of the end-effector, avoiding any obstacle. Trajectory generation consists in parameterizing in time the so obtained curve during the path planning. The resulting time-parameterized trajectory which is commonly called the reference trajectory is obtained primarily in terms of the coordinates in the workspace. Then, following method of inverse kinematics, a time-parameterized trajectory for the joint coordinates, joint space to workspace coordinates with the help of forward kinematics are obtained. Many methods have been suggested for inverse and forward kinematics solution for Puma560 [4-8] taking care of joint singularity. Control design requires exact modeling of system dynamics. The system mathematical model is obtained by analytical or experimental techniques. Tremendous effort has been put in for identification of Puma 560 dynamics [9]. Control of Robot manipulator is a mature yet fruitful area of research, development and manufacturing. The simplest controller for robotic manipulator is PID controller [10-11] and is being widely used in industries. One of the most important factors in designing PID controller is the tuning of PID parameters which is basically an optimization problem. Traditional derivative-based optimization methods are fast and accurate for many types of optimization problems. These methods are designed to solve continuous and differentiable minimization problems, as they use derivatives to determine the direction of descent. However, using derivatives is often ineffective with discontinuous, nondifferentiable or stochastic objective functions. For nonsmooth problem, methods such as the genetic algorithm or the more recently developed pattern search algorithm are effective alternatives.

Genetic algorithms are stochastic search algorithm inspired by the principle of natural selection and natural genetics. Genetic Algorithm has considerably broadened the scope of optimization in engineering [12-13]. Simulated Annealing (SA) is motivated by an analogy to annealing in solids. The algorithm simulates the cooling process by gradually lowering the temperature of the system until it converges to a steady, frozen state. SA's major advantage over other methods is its ability to avoid becoming trapped at local minima. The algorithm employs a random search, which not only accepts changes that decrease objective function, but also some changes that increase it with some probability [14-15]. Generalized pattern search (GPS) algorithms are derivative free methods for the minimization of smooth functions, possibly with linear inequality constraints [16-18]. Generalized pattern search algorithms for unconstrained or linearly constrained minimization generate a sequence of iterates with non-increasing objective function values. Iteration is divided into two phases: an optional search and a local poll.

Sufian Ashraf Mazhari is working in GS E&C Gurgaon , India (Email: sufian.ashraf@gmail.com)
Surendra Kumar is Assistant Professor in the Department of Electrical Engineering, Indian Institute of Technology, Roorkee, Uttrakhand 247667, India (Email: surendra_iitr@yahoo.com)

II. MODELLING OF MANIPULATOR DYNAMIC AND KINEMATICS

The dynamics of an n-link robotic manipulator is characterized by a set of highly nonlinear and strongly coupled second order differential equation.

$$D(\theta)\ddot{\theta} + C(\theta,\dot{\theta}) + G(\theta) + F(\dot{\theta}) = \tau \quad (1)$$

where $D(\theta)$ is the nxn inertial matrix, $C(\theta,\dot{\theta})$ is the nx1 vector of centrifugal forces, $G(\theta)$ is the nx1 vector of gravity loading, $F(\dot{\theta})$ is nx1 vector of friction term. $\theta,\dot{\theta}$ and $\ddot{\theta}$ are nx1 vector for joint angular position, velocity and acceleration, τ is nx1 joint torque vector. D, C, G, F are very complicated function of θ and $\dot{\theta}$. The dynamic parameters of Puma 560 have been taken from [9]. Puma 560 joint actuators are DC servo motors with armature voltage as control input. The motor is connected to manipulator links through gear where the Robot dynamics appears as dynamic load. The dynamics of DC motor can be represented as

$$E_a = E_b + L\frac{dI}{dt} + RI \quad (2)$$

$$E_b = K_e N\Omega \quad (3)$$

$$I = \frac{E_a - K_e N\Omega}{Ls + R} \quad (4)$$

$$\tau = K_m I \quad (5)$$

where E_a is the armature voltage, E_b the Back e.m.f, L and R are inductance and reactance of armature windings respectively, I is the armature current, N is gear ratio, K_e is the back e.m.f constant, K_m is motor constant and Ω is load angular velocity. Actuator data of puma 560 Robot [20] is given in appendix A. Puma 560 contains PID controller with controller output u_i as

$$u_i = k_p e + k_i \int e dt + k_d \dot{e}$$

The transformation between the joint space and the Cartesian space of the robot is very important since robots are controlled in the joint space, whereas tasks are defined and object manipulated in the Cartesian space. The kinematics problem deals with the analytical study of the relation between these two spaces. The direct kinematics defined as the transformation from the joint space to the Cartesian space and the inverse kinematics defined as the transformation from the Cartesian space to the joint space. While modeling the kinematics of manipulator, arm singularity and configuration must be checked. Many methods have been proposed for better and feasible solution of manipulator kinematic problems [8-10]. The Forward and Inverse kinematic equations have been modeled [11] and are given in appendix A. Control system diagram of Puma 560 is shown in Fig.1 which consists of desired Cartesian space trajectory T, inverse kinematics block I, PID controller, servo motor M, dynamics D and forward kinematics Block F.

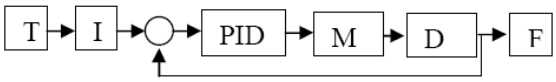

Fig. 1 Block diagram representation of Puma 560 control system

The Matlab/Simulink Model of dynamics, forward kinematics and Inverse kinematics are designed in such a way that all operations are matrix based because Matlab incorporates LAPACK to increase the speed and complexity of matrix computation. Simulink model of Forward dynamics is shown in Fig.2 and simulink model of complete system is shown in Fig.3.

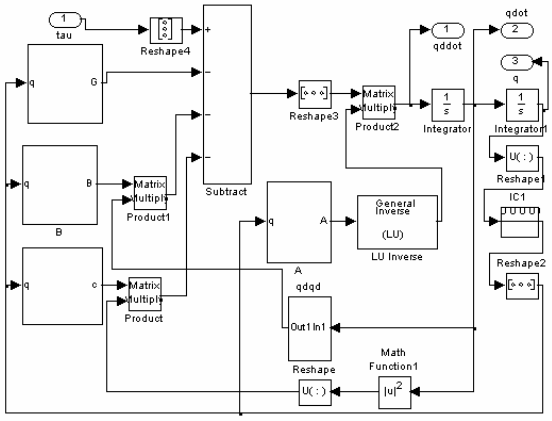

Fig. 2 Simulink diagram of PUMA560 dynamics

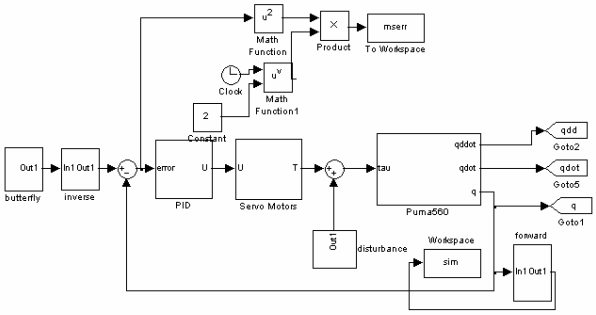

Fig. 3 Simulink diagram of PUMA560 system

III. TUNING METHODS

A. GA Tuning of PID

The main advantage of using GA for tuning PID controller is its adaptability to adopt any constraints. Determining the upper and lower bound of PID parameters for GA based tuning is difficult and needs a large number of experiments. In GA tuning approach conventional method based values can be used to give upper and lower bound of parameters value as

$$kp_{ij} \in [\frac{kp_0}{3}, 3kp_0], ki_{ij} \in [\frac{ki_0}{3}, 3ki_0], kd_{ij} \in [\frac{kd_0}{3}, 3kd_0]$$

where j=1,2…..6. kp_{ij} , ki_{ij} and kd_{ij} are proportional gain, integral gain and derivative gain of joint j at ith iteration respectively. $[kp_0, ki_0, kd_0]$ are values of PID parameters taken from [20].Puma 560 contains independent controller for each joint. For six joint controllers, there are 18 values for PID parameters. By taking an 18 variable string as $[k_{p1}...k_{p6}..k_{i1}....k_{i6}..k_{p1}....k_{d6}]$ for GA, an optimal value can be searched. The method of tuning PID parameters using GA is based upon minimizing the integral time squared error of joints. If $q_d(k)$ is desired trajectory and $q(k)$ is output trajectory then error $e(k)$ is

$$e(k) = q_d(k) - q(k) \tag{6}$$

$$ITSE = \sum_{j=1}^{6}\sum_{k=1}^{n} e^2(ki).ki^2 \tag{7}$$

where e(ki) is the system error at k^{th} sampling instant for j^{th} joint. GA acts as a controller which modifies the set of parameters

$$\{K_{pj}\}_{j\in 1;p}, \{T_{ij}\}_{j\in 1;p}, \{T_{dj}\}_{j\in 1;p}$$

of the j[th] population which consist of P individual parameters of control system .This cycle is repeated until convergence criteria is met. In the evaluation step of GA, a simulation is performed for each u_i .Parameters used for simulation of GA are Population size=60, iteration=500, Crossover probability=0.8, Mutation probability =0.01

B. Simulated Annealing

The algorithm employs a random search, which not only accepts changes that decrease objective function, *ITSE*, but also some changes that increase it. The latter are accepted with a probability .In each step of algorithm, a particle is given a small random displacement and the resulting change in the energy of the system $\delta ITSE$ is computed. For $\delta ITSE \le 0$, the displacement is accepted and for $\delta ITSE > 0$ displacement is accepted with probability. The probability that the configuration is accepted is given in equation (8). Certain number of iterations is carried out at each temperature and then the temperature is decreased. This is repeated until the system freezes into a steady state.

$$P_i = \exp(-\frac{\delta ITSE_i}{T_i}) > r_i \tag{8}$$

where $\delta ITSE$ = the change in objective function

$T = [k_{pj}, k_{ij}, k_{dj}]$ Control Parameters (Temperatures)

r = a random number uniformly distributed between 0 and 1.

The probability of accepting a worse move is a function of both the temperature of the system and change in the objective function. As the temperature of the system decreases, the probability of accepting a worse move is decreased. If the temperature is zero, then only better moves will be accepted.

C. Generalized Pattern Search

Generalized Pattern search (GPS) is a sub-class of direct search algorithms, which involve the direct comparision of objective function values and information about the gradient of the objective function. As opposed to more traditional optimization methods that use information about the gradient or higher derivative to search for an optimal point, a direct search algorithm searches a set of points around the current point, looking for one where the value of the objective function is lower than the value at the current point. The direction set and a step length parameter define a conceptual mesh centered about the current iterate. Trial points are selected from the mesh, evaluated, and compared to the incumbent in order to select the next iterate. If an improvement is found among the trial points, the iteration is declared successful and the mesh is retained or coarsened; otherwise, the mesh is refined and a new set of trial points is constructed .The key to generating the mesh is the definition of the direction set. This set must be sufficiently rich to ensure that function value is decreasing along at least one of the directions. Formulation Problem for tuning using GPS is

$$\text{Minimize} \quad ITSE = \sum_{j=1}^{6}\sum_{k=1}^{n} e^2(ki).ki^2$$

$$[kp_0 ki_0 kd_0]/3 \le [kp_{ij} ki_{ij} kd_{ij}] \le 3*[kp_0 ki_0 kd_0]$$

where $[kp_{ij} ki_{ij} kd_{ij}] \in R$ j=1,2…..6. Where kp_{ij} , ki_{ij} and kd_{ij} are proportional gain, integral gain and derivative gain at i[th] iteration of joint j. respectively. Parameters used for simulation of GPS are initial mesh size=10, maximum mess size=100.

D. Hybrid Search

GA can reach the region near an optimum point relatively quickly, but it can take many function evaluations to achieve convergence. GA is not good at identifying the optimal value of a chromosome for a problem but very quickly identifying the regions where those optima lie. Unlike other search and optimization techniques, GA promises convergence but not optimality. If we run the GA several times, it will converge each time, possibly at different optimal chromosomes. This implies that the choice of stopping GA iteration is not well-defined. A commonly used technique is to run GA for a small number of generations to get near an optimum point. Then the solution from GA is used as an initial point for another optimization solver that is faster and more efficient for local search. Typically, GA is coupled with a local search mechanism to find the optimal chromosome in a region. In this paper SA and GPS are being used for hybrid search with GA. Initially GA is being used to confine the search space with 50 iterations and population size 60.Values returned from GA are being used with SA and GPS for local search.

IV. Results

A. Tuning of controller

PID controller is tuned with GA for 500 iteration. Convergence time of GA is very high. Graph for fitness value tuned with GA is shown in Fig.4.

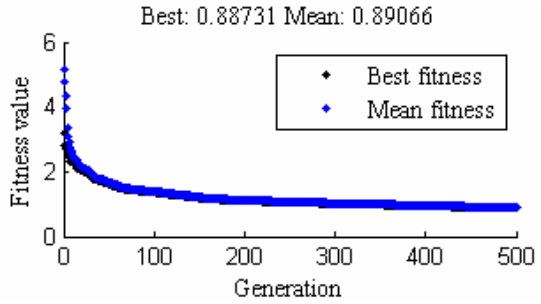

Fig.4 Best and Mean fitness curve of GA

Performance of tuning PID controller with SA is found better compared to tuning using GA. Graph for fitness value tuned with SA is shown in Fig.5.Time taken to tune PID using SA is less compared to GA.

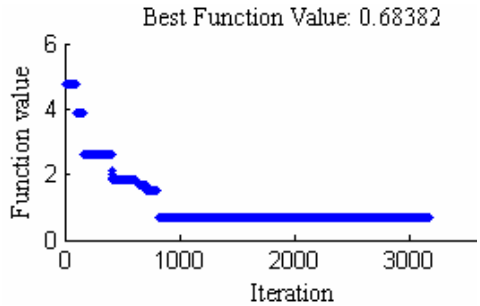

Fig.5 Best fitness curve of SA

GPS is used to tune controller with very less iteration and less time, it is showing better performance compared to GA and SA. Graph for fitness value tuned with GPS is shown in Fig.6.Hybrid GA search technique using GPS and SA is used to tune PID parameters in search of performance better than that of returned by GPS. Convergence curve of both hybrid techniques are shown in Fig.7.a and Fig.7.b It clearly reflects that GA with GPS give better result compared to other methods. PID parameters value returned with GA-GPS tuning is shown mentioned in Table. I

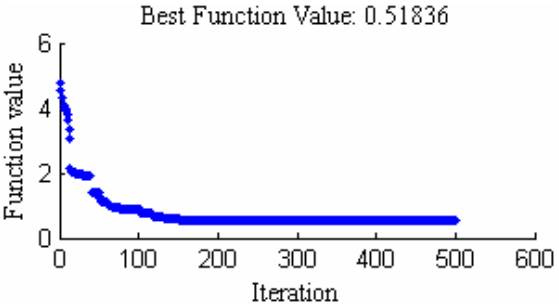

Fig.6 Best fitness curve of GPS

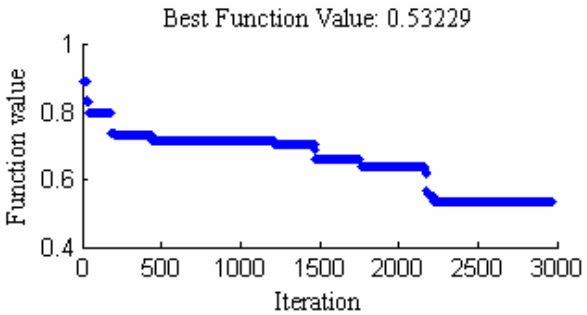

Fig.7.a Best fitness curve of GA-SA

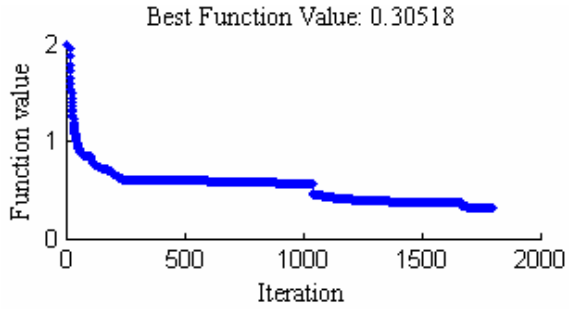

Fig.7.b Best fitness curve of GA-GPS

TABLE I
GA-GPS TUNED PID PARAMETER VALUES

Joint	*kp*	*ki*	*kd*
1	644.31	1978.01	14.03
2	738.38	1803.18	21.39
3	659.89	2457.19	10.13
4	1320.55	7179.15	15.13
5	1312.88	7289.25	15.19
6	1344.77	7971.69	15.19

B. Testing of controller performance

Results of all the tuning methods are tested in terms of ITSE in Cartesian space and ISE in joint space. The desired Cartesian space trajectories taken for testing controller are circle and butterfly (x-y plane trajectory). The Cartesian space points x_d, y_d are converted into joint space angle with inverse kinematics method and motors are controlled in joint space. Joint space error ITSE is calculated and for corresponding joint angle, cartesian space points *(x,y)* are calculated.

Integral square errors ISEX and ISEY are calculated to compare controller performance.

$$ISEX = \sum_{k=1}^{n} (x_d(k) - x(k))^2 \quad (9)$$

$$ISEY = \sum_{k=1}^{n} (y_d(k) - y(k))^2 \quad (10)$$

Where $[x_d(k), y_d(k)]$ and $[x(k), y(k)]$ are desired and output cartesian space points at k^{th} sampling instant. Parametric equation of the circle the taken is

$$x_d = 0.15\cos(2u) + 0.35 \quad (11)$$

$$y_d = 0.15\sin(2u) + 0.2 \quad (12)$$

Parametric equation of Butterfly trajectory is

$$x_d = 0.02\cos(t)(e^{\cos t} - 2 .\cos(4t) - \sin^5(t/12)) + 0.35 \quad (13)$$

$$y_d = 0.02\sin(t)(e^{\cos t} - 2 .\cos(4t) - \sin^5(t/12)) + 0.2 \quad (14)$$

For checking the robustness of controller a disturbance torque D is applied

$$D = 7.5\sin(4.3575t) + 3.5 .\sin(9.825) + 3.5\sin(2.7075) - 4.5$$

The sampling time of system is 1ms. Initial angles of all six joints are set at zero, however both circular and butterfly joint space trajectory is starting from non zero value which gives step input behavior and it is clearly reflected in Fig.13.a to Fig.13.c .The angle of all six joint for tracking circle with and without disturbance are shown in Fig.9.a to Fig.9.d and corresponding ITSE, ISEX and ISEY are shown in Table. II and Table.III. Very large error in joint space is found with Untuned PID controller for tracing circle and butterfly with and without disturbance, which clearly shows fine tuning of controller is desired Fig.13.a shows that untuned PID controller fails to track a typical butterfly trajectory. Desired input, actual output joint space trajectory for tracking butterfly with and without disturbance of untuned PID controller are shown in Fig.10.a to Fig.10.d. The controller is tuned with GA, SA and GPS.GA tuned PID controller is showing better performance compared to untuned PID controller but SA is showing better performance compared to GA.GPS technique is showing better result compared to GA and SA. The performance of all the tuning method is compared in Table II and III. Comparing GA convergence curve for 500 iterations Fig.4 and 50 iterations Fig.8, it shows that after 500 iteration GA is converging with fitness value 0.88731 while reaching 0.94233 in less than 50 iteration only. This gives a direction to move towards hybrid GA search technique. GA-GPS and GA-SA hybrid techniques are investigated and it is found that GA-GPS hybrid technique is best among all techniques. GA-GPS is tested for desired circular and butterfly trajectories with and without disturbance in Fig.11 and Fig.12.Untuned PID and GA, and GA-GPS tuned PID used to track butterfly trajectory with disturbance is shown in Fig.13.a to Fig.13.c which clearly reflects difference between untuned and tuned controller. Corresponding change in desired and actual output trajectory dx and dy is shown in Fig.14.a to Fig.14.c

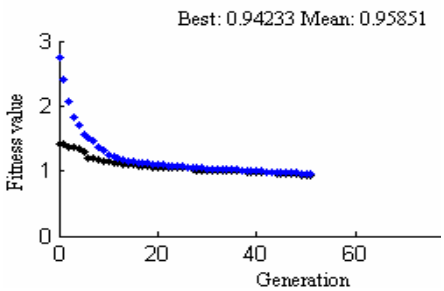

Fig. 8 GA fitness curve for 50 iterations

TABLE II
ITSE FOR ALL METHODS

ISE	Circle		Butterfly	
	Without D	With D	Without D	With D
untuned	1.7552	2.4988	42.8599	1.6264e+3
GA	1.6986	1.0123	17.5934	1..341e+3
SA	0.4223	0.8931	12.3256	1.145e+3
GPS	0.2587	0.4193	2.7545	967.0231
GA-SA	0.3809	1.0315	7.8821	1.0095e+3
GA-GPS	0.1535	0.2637	4.1985	340.6403

TABLE III
ISEX AND ISEY FOR ALL METHODS

ISEX	Circle		Butterfly	
ISEY	without *D*	with *D*	without *D*	with *D*
Untuned	0.6125	0.6176	0.7390	0.7402
	0.1631	0.1684	0.1430	0.1852
GA	0.7745	0.7833	0.2060	0.2080
	0.1281	0.1310	0.1050	0.6370
SA	0.4936	0.4960	0.1751	0.2792
	0.0888	0.0923	0.1011	0.1247
GPS	0.5357	0.5363	0.3077	0.3094
	0.1118	0.1134	0.0958	0.1063
GA-SA	0.5183	0.5206	0.4470	0.4519
	0.0944	0.0949	0.0860	0.0925
GA-GPS	0.2272	0.2331	0.2406	0.2490
	0.0906	0.0920	0.0804	0.1009

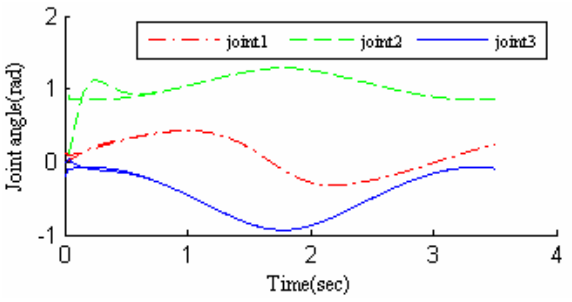

Fig. 9.a Desired and actual joint angles of circular trajectory for untuned PID without disturbance

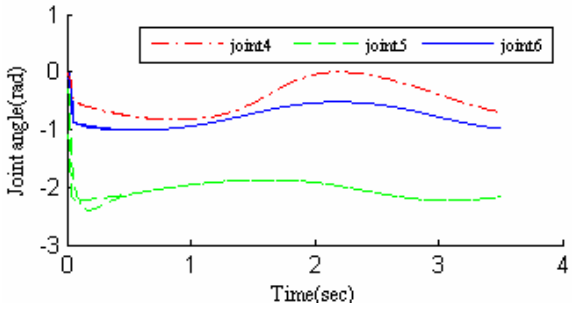

Fig. 9.b Desired and actual joint angles of circular trajectory for untuned PID without disturbance

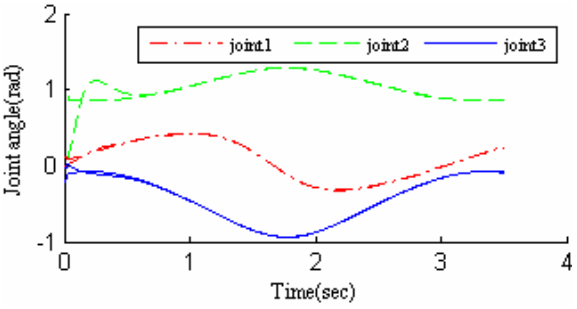

Fig .9.c Joint Desired and actual joint angles of circular trajectory for untuned PID with disturbance

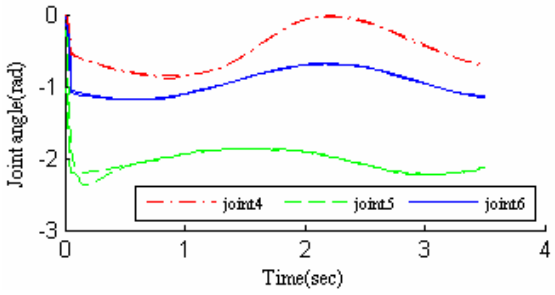

Fig. 9.d Joint Desired and actual joint angles of circular trajectory for untuned PID with disturbance

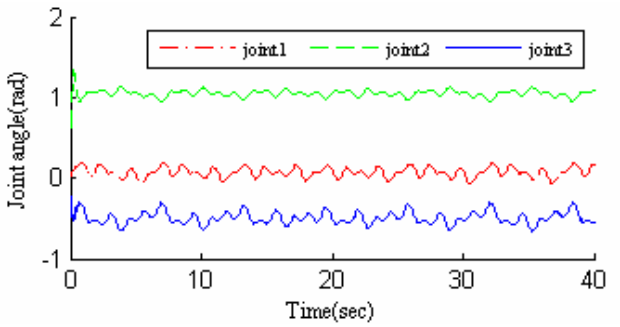

Fig. 10.a Desired and actual joint angles of butterfly trajectory for untuned PID without disturbance

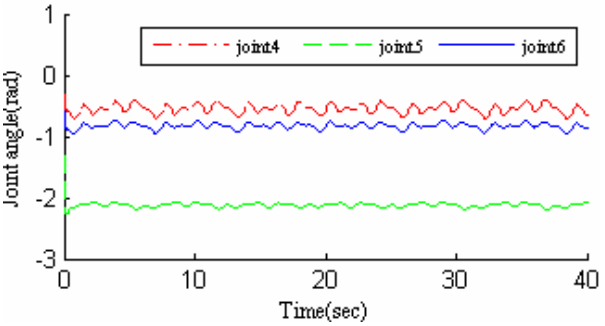

Fig. 10.b Joint Desired and actual joint angles of butterfly trajectory for untuned PID without disturbance

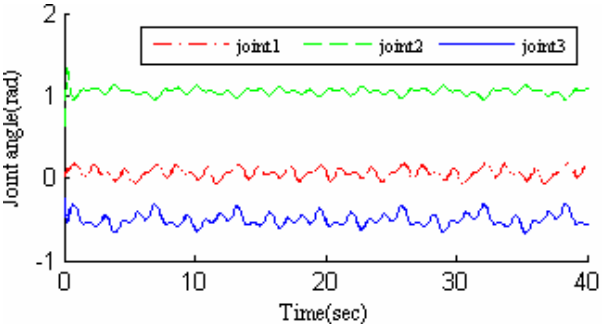

Fig. 10.c Desired and actual joint angles of butterfly trajectory for untuned PID withdisturbance

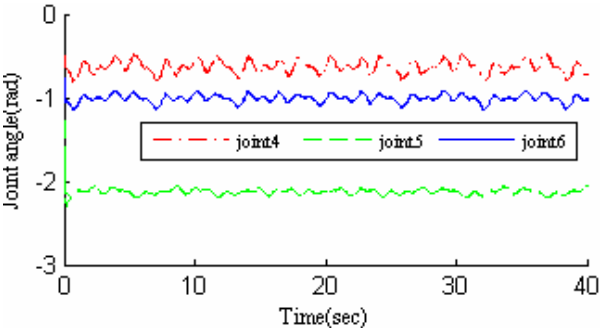

Fig. 10.d Desired and actual joint angles of butterfly trajectory for untuned PID with disturbance

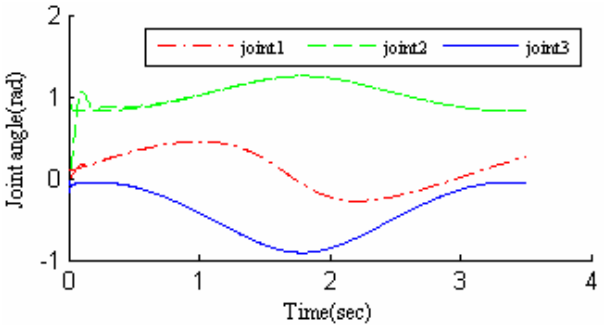

Fig. 11.a Desired and actual joint angles of circular trajectory for GA-GPS tuned PID without disturbance

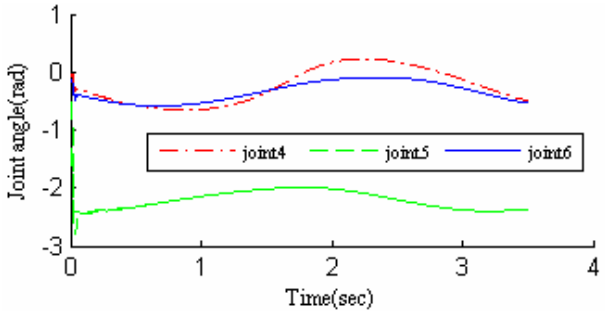

Fig. 11.b Desired and actual joint angles of circular trajectory for GA-GPS tuned PID without disturbance

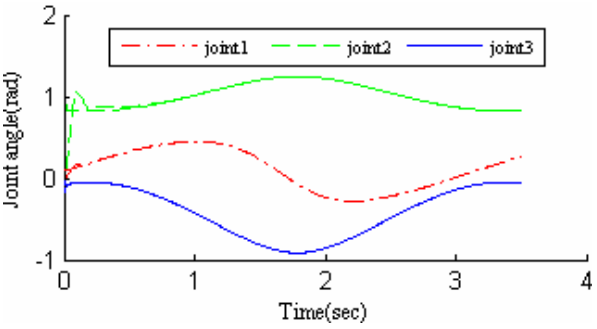

Fig. 11.c Desired and actual joint angles of circular trajectory for GA-GPS tuned PID with disturbance

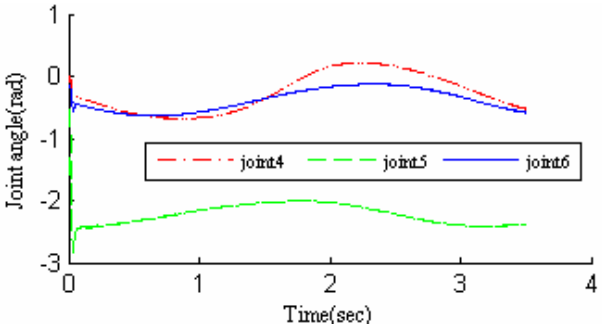

Fig. 11.d Desired and actual joint angles of circular trajectory for GA-GPS tuned PID with disturbance

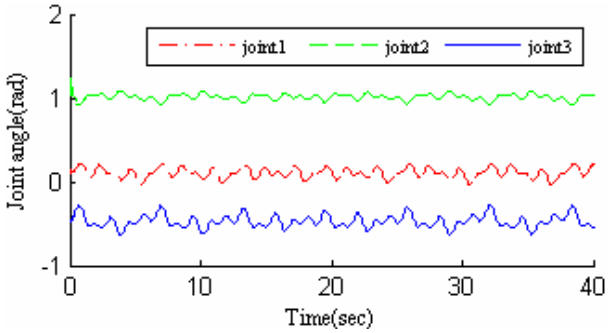

Fig. 12.a Desired and actual joint angles of butterfly trajectory for GA-GPS tuned PID without disturbance

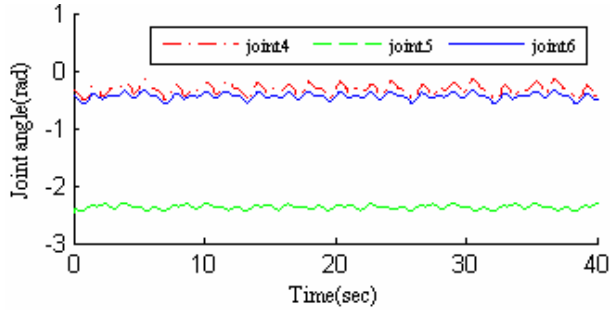

Fig. 12.b Desired and actual joint angles of butterfly trajectory for GA-GPS tuned PID without disturbance

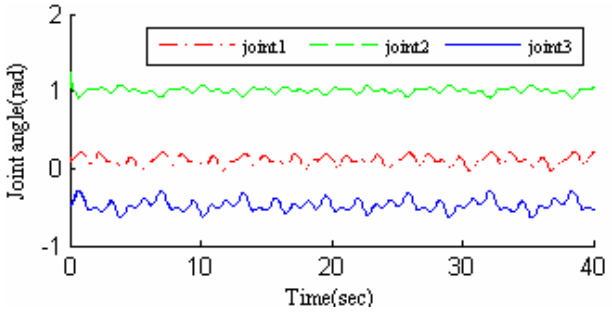

Fig. 12.c Desired and actual joint angles of butterfly trajectory for GA-GPS tuned PID with disturbance

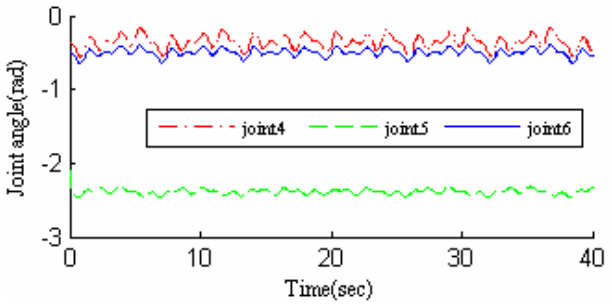

Fig. 12.d Desired and actual joint angles of butterfly trajectory for GA-GPS tuned PID with disturbance

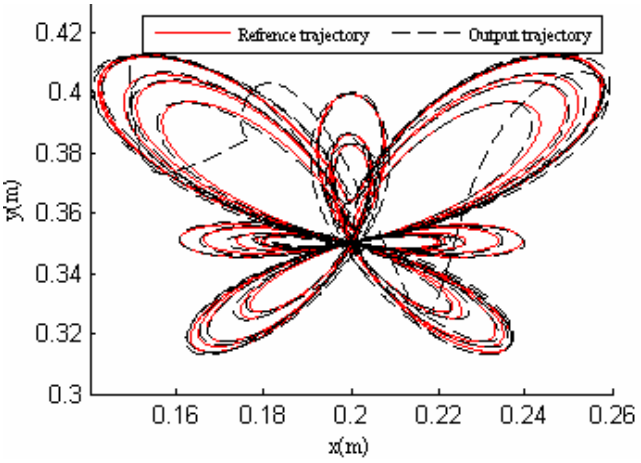

Fig. 13.a Desired and actual output butterfly trajectory with untuned PID

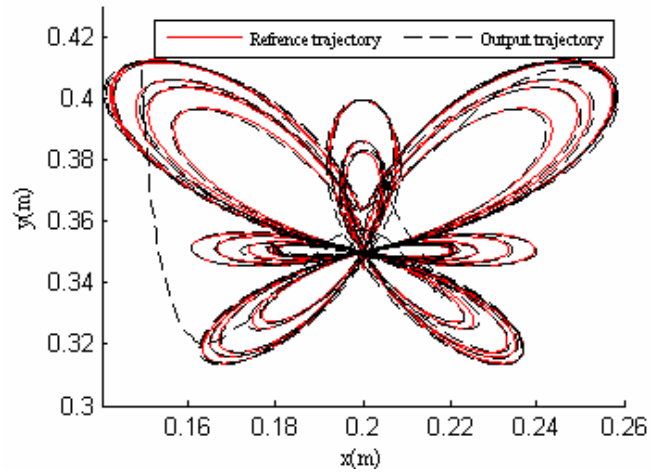

Fig. 13.b Desired and actual output butterfly trajectory with GA tuned PID

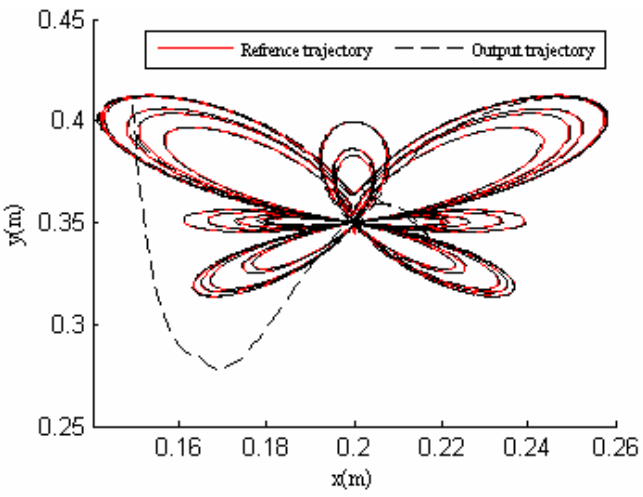

Fig. 13.c Desired and actual output butterfly trajectory with GA-GPS tuned PID

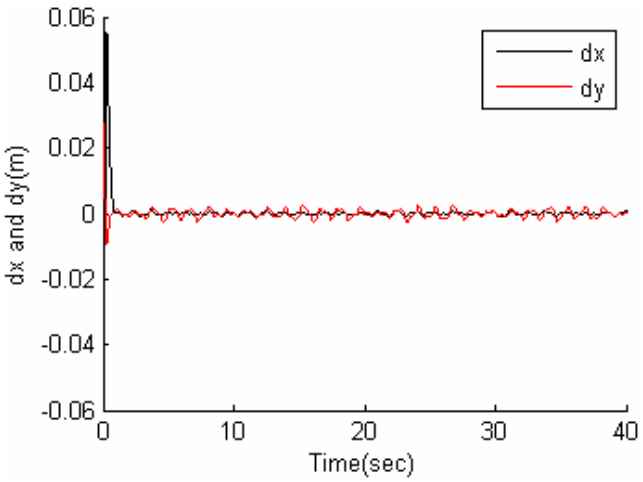

Fig. 14.a Cartesian space error for butterfly trajectory with untuned PID

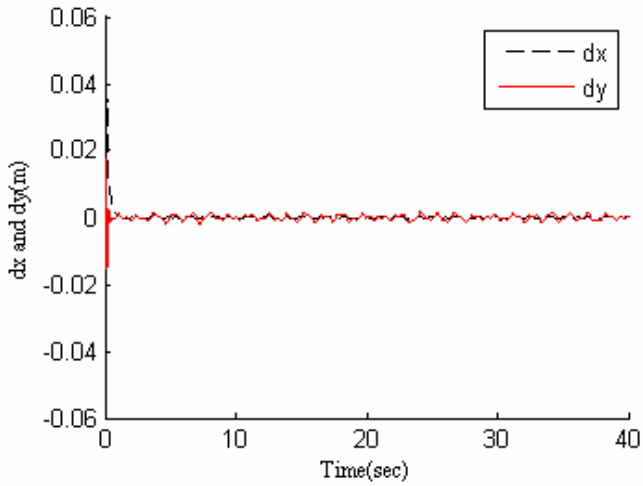

Fig. 14.b Cartesian space error for butterfly trajectory with GA tuned PID

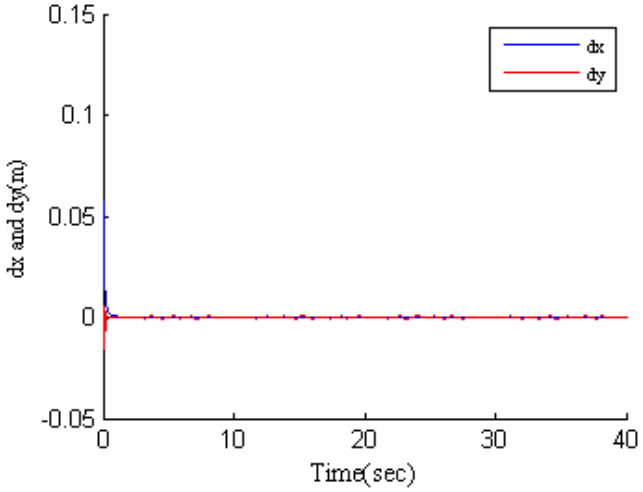

Fig. 14.c Cartesian space error for butterfly trajectory with GA-GPS tuned PID

V. Conclusion

In this paper a very fast forward kinematics, inverse kinematics and forward dynamics of the PUMA560 is modeled on Matlab/Simulink taking care of Matlab matrix calculation capability. The simulation time is found to be very low as compared to latest version of robotic toolbox [20].Tuning of PID controller was investigated using GA, SA, GPS and Hybrid GA techniques. It can be concluded from the results that GA-GPS is showing best results as compared to GA and SA .However looking at the convergence characteristics of GA , it is found that, GA is reaching near search space very fast compared to other methods. Taking care of this hybrid search technique is investigated which consists of running GA for few iterations to confine the search space and then using SA or GPS for further searching. All the methods were compared in term of ISE in Cartesian space and ITSE in joint space with and without disturbance. It was found that GA with GPS is showing best performance and robustness.

References

[1] Chul-Goo Kang Online "Trajectory Planning for a PUMA Robot," *International Journal of Precision Engineering and Manufacturing, vol.* 8, vo.4, pp.16-21, Oct. 2007.
[2] Francisco Valero and Vicente Mata, Antonio Besa "Trajectory planning in workspaces with obstacles taking into account the dynamic robot behavior," *Journal of Mechanism and Machine Theory,* vol. 41, issue 5, pp. 525-536,May. 2006.
[3] Chia-Yu E. Wang, Wojciech K. Timoszyk, and James E. Bobrow "Payload Maximization for Open Chained Manipulators: Finding Weightlifting Motions for a Puma 762 Robot," *IEEE Transactions on Robotics and Automation*, vol. 17, no. 2, Apr. 2001.
[4] Shadia Elgazzar "Efficient Kinematic Transformations for the PUMA 560 Robot," *IEEE Journal Of Robotics And Automation,* vol. Ra-1, no. 3, Sept. 1985.
[5] Said M. Megahed "Inverse Kinematics of Spherical Wrist Robot Arms Analysis and Simulation," *Journal of Intelligent and Robotic Systems*, vol. 5, pp. 211-227, 1992.
[6] Jean Cote, C1ement M. Gosselin and Denis Laurendeau "Generalized Inverse Kinematic Functions for the Puma Manipulators," *IEEE Transactions on Robotics and Automation*, vol. 1 I, no. 3, Jun. 1995.
[7] Fan-Tien Cheng, Tzung-Liang Hour, York-Yin Sun, and Tsing-Hua Chen "Study and Resolution of Singularities for a 6-Dof Puma Manipulator "*IEEE Transactions on Systems, Man, and Cybernetics—Part B: Cybernetics,* vol. 27, no. 2, Apr. 1997.
[8] Frederic Chapelle and Philippe Bidaud, "Closed form solutions for inverse kinematics approximation of general 6R manipulators," *Journal of Mechanism and Machine Theory,* vol. 39, issue. 3, pp. 323-338, Mar 2004.
[9] Brian Armstrong, Oussama Khatib, Joel Burdick,"The explicit dynamic Model and Inertial Parameters of the Puma 560 Arms," *IEEE International conference on Robotics and Automation*, vol. 2 ,pp. 1608-1613, May 1994.
[10] John T. Wen and Steve H. Murphy, "PID control for robot manipulators", Rensselaer Polytechnic Institute, CIRSSE Document #54, May 1990.
[11] R.Kelly, V.santibanez A.Loria, *Control of Robot Manipulators in joint space*, Springer Advanced Textbooks in Control and Signal Processing, series 2005
[12] David A.Coley, *Introduction to Genetic Algorithm for scientist and engineer*, World scientific Publishing 1999
[13] D.P.KWOK, Fang Sheng "Genetic algorithm and simulated annealing for optimal robot am PID control," *IEEE conference on evolutionary computing,* vol. 2, pp. 707 – 713, June 1994.
[14] S. Kirkpatrick, C. D. Gelatt, Jr., M. P. Vecchi, "Optimization by simulated annealing," *Science*, vol. 220, no. 459813, May 1983.
[15] R.A.Rutenber," Simulated Annealing Algorithms: An Overview," *IEEE Circuits and Devices*, pp. 19-26,Jan. 1989.
[16] Robert Hooke, T. A. Jeeves, "Direct Search Solution of Numerical and Statistical Problems," *Journal of the ACM*, vol. 8, issue 2 pp: 212 - 229, April 1961.
[17] Robert Michael Lewis and Virginia Torczon, "Pattern Search Methods For Linearly Constrained Minimization," Institute for Computer Applications in Science and Engineering, Report no.98-3, 1998.
[18] LiYing Liu, XueSheng Zhang "Generalized pattern search methods for linearly equality constrained optimization problems," *Applied Mathematics and Computation*, vol. 181, Issue 1, pp 527-535, Oct. 2006.
[19] T.J.Tarn, A.K.Bejczy, G.T.Marth, A.K.Ramadorai "Performance Comparison of Four Manipulator Servo Schemes" *IEEE control system magazine,* vol. 3 issue 1, Feb 1993.
[20] P. Corke, "A robotics toolbox for MATLAB," IEEE Robotics and Automation Magazine, vol. 3, pp. 24–32, Mar 1996.
[21] Naganna G.E and Surendra Kumar," Conventional and intelligent controllers for robotic manipulator," IEEE International conference on industrial technology, pp. 424-428, Dec 2006.

Sufian Ashraf Mazhari was born in India and received his Bachelor degree (Electrical Engineering) from Aligarh Muslim University, Aligarh India, in 2005 and M.Tech degree in Electrical Engineering from Indian Institute of Technology, Roorkee India.Currently is working with GS E & C ,Gurgaon ,India. His area of research includes Robotic Control, Optimization and Computer vision system.

Surendra Kumar (M'07) received B.E (Electrical), M.E(System Engineering and Operations Research) and Ph.D. Electrical in 1969,1971 and 1982 respectively in India. He joined the Department of Electrical Engineering, Indian Institute of Technology Roorkee, India as lecturer in 1972 .He has 36 year of teaching and research experience. Presently he is Assistant Professor .He has been on teaching assignment to University of Technology, Baghdad, IRAQ during 1987-1989.He is member IEEE, Fellow of Institution of Engineers India and member of ISTE India. His area of research interest is mainly Control, Optimization, AI application to Robotic Control and Fuzzy Reliability.

VI. Appendix. A

Abbreviation used:

$c_i = \cos(\theta_i), s_i = \sin(\theta_i), c_{ij} = \cos(\theta_i + \theta_j)$

$s_{ij} = \sin(\theta_i + \theta_j), s_{i,-j} = \sin(\theta_i - \theta_j)$

The arm configuration parameters of Puma 560 k_1, k_2 and k_3 are defined as

$$k_1 = \begin{cases} +1, \text{ lefty} \\ -1, \text{ righty} \end{cases}$$

$$k_2 = \begin{cases} +1, \text{ elbow up} \\ -1, \text{ elbow down} \end{cases}$$

$$k_3 = \begin{cases} +1, \text{ no flip} \\ -1, \text{ flip} \end{cases}$$

The parameters are k_1, k_2 and k_3 are used to find inverse kinematics solution. However in the case of a known set of

joint angles, as in the case of the direct kinematics, these parameters can be computed.

Forward kinematics:

The problem is defined as given the joint angles vector, find the Cartesian position/orientation vector *R,* and the arm configuration parameters k_1, k_2, k_3

The orientation angles r_θ, r_ψ and r_ρ are defined as

$\cos(r_\vartheta) = \mathrm{c}_{23}\,\mathrm{c}_5 - \mathrm{s}_{23}\,\mathrm{s}_5\,\mathrm{c}_5$

$r_\psi = \theta_6 + a\tan 2[\mathrm{s}_{23}\,\mathrm{s}_4, \mathrm{s}_5\,\mathrm{c}_{23} + \mathrm{s}_{23}\,\mathrm{c}_5\,\mathrm{c}_4]$

$r_\rho = \theta_1 + a\tan 2[\mathrm{s}_5\,\mathrm{s}_4, \mathrm{c}_5\,\mathrm{s}_{23} + \mathrm{c}_{23}\,\mathrm{s}_5\,\mathrm{c}_4]$

Where atan2 (x, y) is four-quadrant version $\tan^{-1}(x/y)$. As $|\cos(r_\theta)| \to 1$ the accuracy of equations deteriorates because $|\cos(r_\theta)| = 1$ is a singular point. If $\sin(r_\theta) \approx 0$, r_θ is set to zero or π depending upon sign of $\cos(r_\theta)$. Value of r_ψ is set to zero and r_ρ is calculated using

$r_\rho = \theta_1 + a\tan 2[2\,\mathrm{s}_{46}/[c_{23} + c_{r_\theta}], \mathrm{c}_{46}]$

Position vector *R, is defined as*

$r_x = -w_b s_1 - d c_1 - l_4 s_{r_\theta} s_{r_\rho}$

$r_y = w_b c_1 - d s_1 + l_4 s_{r_\theta} c_{r_\rho}$

$r_z = w_a + l_1 + l_4 c_{r_\theta}$

Where

$w_a = l_2 c_2 + l_3 c_{23}$,

$w_b = l_2 s_2 + l_3 s_{23}$,

The arm configuration is determined by evaluating parameters k_1, k_2, k_3 .If $w_b \geq 0$ then the arm is lefty and $k_1 = +1$, but if $w_b < 0$, then the arm is righty and $k_1 = -1$. If $k_1\theta_3 \geq 0$ then the arm is elbow up and $k_2 = +1$ else $k_2 = -1$ .if $\theta_5 \geq 0$ then a no-flip solution exists and $k_3 = +1$ but if $\theta_5 < 0$ then a flip solution exist and $k_3 = -1$

Inverse kinematics:

The problem is defined as given Cartesian position/orientation vector *R,* and the arm configuration parameters k_1, k_2, k_3 find the joint angles vector.

Joint angles $\theta_1, \theta_2 ....., \theta_6$ are given as

$\theta_1 = a\tan 2(-k_1 w_{1x}, k_1 w_{1y}) - k_1 a\tan 2(d, l)$

Where

$w_1 = (r_x + l_4 s_{r_\theta} s_{r_\rho})i + (r_y - l_4 s_{r_\theta} c_{r_\rho})j + (r_z - l_4 c_{r_\theta})k$ Singular point exists if $w_{1x} = w_{1y}$. However considering the arm geometry this condition is never satisfied

$\cos(\theta_3^{'}) = \dfrac{n^2 - l_2^{\,2} - l_3^{\,2}}{2 l_2 l_3}$

$\theta_3 = k_1 k_2 \theta_3^{'} + \delta$

$\mathrm{c}_5 = c_{23} c_{r_\theta} + s_{23} s_{r_\theta} \cos(r_\rho - \theta_1)$

if $s_5 > \xi$ $\theta_4 = a\tan 2[s_{23} s_{r_\rho,-1}, c_{23} s_{r_\theta} c_{r_\rho,-1} - c_{r_\theta} s_{23}]$

$\theta_6 = r_\psi - \beta$

Where

$\beta = a\tan 2[s_{23} s_{r\rho,-1}, s_{r_\theta} c_{23} - c_{r_\theta} s_{23} c_{r\rho,-1}]$

Manipulator loses a degree of freedom when two joint axes become collinear. This is case when $\sin(\theta_5) = 0$ and, consequently θ_4 and θ_6 become linearly dependent. The accuracy of θ_4 and β deteriorates as $\sin(\theta_5) \to 0$ and they break down completely if $\sin(\theta_5) = 0$. Therefore for some value of ξ and for $\sin(\theta_5) \leq \xi$ better value of θ_4 and θ_6 can be obtained using the equation

$\theta_4 - \beta = a\tan 2\left\{0.5 s_{r_\rho,-1}[c_{23} + c_{r_\theta}], c_{r_\rho,-1}\right\}$

In the case of a no-flip condition, that is $k_3 = 1$ the wrist angles are obtained from the above equations. If, however, $k_3 = -1$, the flip solution becomes $(\theta_4 + \pi, -\theta_5, \theta_6 + \pi)$

$l_1 = .672m, l_2 = .432m, l_3 = .433m$

$l_4 = .056m, d = .149m$(offset in arm due to shoulder offset and elbow offset), $\delta = 2.72^\circ$

Actuator data of Puma 560 robot

Motor	R	L	K_e	K_m	N
1	1.6	0.0048	0.19	0.2611	62.55
2	1.6	0.0048	0.19	0.2611	107.81
3	1.6	0.0048	0.19	0.2611	53.15
4	3.9	0.0039	0.12	0.0988	76.04
5	3.9	0.0039	0.12	0.0988	71.92
6	3.9	0.0039	0.12	0.0988	76.65