Inverse Acceleration Solution for Robot Manipulators using Harmony Search Algorithm

Hazim Nasir Ghafil
University of Kufa
Engineering collage,
Mechanical Department,
Najaf-Iraq

ABSTRACT

Inverse acceleration problem is very difficult for a serial robot having less than 6 degree of freedom (DOFs), this difficulty is due to the complexity of the inverse Jacobian matrix. For the sake of this problem an approach to solve inverse acceleration of a robot was introduced in this paper, in which harmony search algorithm (HSA) was used to calculate the inverse problem without calculating inverse Jacobian matrix. It is proved that it is applicable by simulation inverse acceleration for a 3-DOF robot. ANSYS 15.0 was used as a simulation software package.

Keywords

Robot; Inverse acceleration solution; Harmony search algorithm

1. INTRODUCTION

One of the most difficult problems encountered in the inverse acceleration Kinematics is the non-square Jacobian matrix which is impossible to be inverted to obtain angular acceleration in joint space. In trajectory planning process, the end-effector has to move over a certain path with a specific value of velocity and acceleration in each segment of the path, so there is a need to transform that acceleration of the tool tip to the joint space i.e. angular acceleration of the actuator. Many researches in field of the inverse Kinematics acceleration, velocity or position had used different methods. One of these methods is the optimization algorithms such as hybrid genetic algorithm [1] and particle swan object PSO [2]. In some applications serial manipulators has to start its motion with acceleration and decelerate until it reach to the stop point, this scenario may be encountered in liquid manipulation like arm prosthesis manipulates a cup of tea, also in applied mechanics, when robot manipulator transforming a large masses, sudden moving or stopping of the arm causes an inertial load that may cause a mechanical failure in the joints or links of the robot arm.

Analytic inverse acceleration is not available for any type of serial robots but forward acceleration problem is easy to solve, in this paper a method to solve the inverse problem for any type of robots by optimizing its forward acceleration equation. The method proposed in this study uses harmony search algorithm for the optimization process because this algorithm continuously refine a given random solution until it reach to the desired one like a Jazz musician when playing random pitch and keep change until he get the best one.. Although the proposed algorithm gives a multi-solutions for a specific point in the task space, the method meet the need and still good enough because it provide inverse acceleration solutions for any type of serial robot manipulator this because (HSA) avoid calculation of the inverse Jacobian matrix and thus no matter how many (DOFs) the robot are.

2. REVIEW

Many researches had interested in field of the inverse Kinematics problem position, velocity and acceleration. Analytical and geometrical approaches was introduced to solve the forward and inverse Kinematics problem for a 6DOF robot manipulator (PUMA)[3], [4]. Iterative method based on Newton-Raphson method for nonlinear equations solution was proposed to solve the inverse problem and overcome the constraints developed by the closed form solution encountered in a class of robots, the method is suitable for any robot have revolute or prismatic joints with any DOFs [5]. Resolved motion rate method presented to control a redundant robot by using {1}-inverse [6]. A driven formula from a closed form solution of a redundant robot manipulators by using Lagrangian multiplier method, the proposed method got ride from the repeatability problem which appear in the resolved-motion method [7]. A modular architecture was developed for inverse Kinematics problem for robot manipulator, this modular was based on a nonlinear equation solver, the solver was based on Banach Fixed-point theorem [8]. Discussion for the Kinematics problem of a hybrid robot manipulator which combines the serial chain with parallelism, also the singularity and the indeterminacy in the inverse problem was discussed [9]. An efficient approach was presented to solve the inverse Jacobian of the wrist partitioned manipulators in case of the differential inverse Kinematics problem, the study used PUMA 560 as a case study and the inverse Jacobian matrix was formed on the basis of the generated simplified equations [10]. A proposed closed-loop algorithm to solve the inverse Kinematics without calculating Jacobian matrix, avoidance of the singularities that may cause numerical instabilities was discussed in [11]. Numerical solution for the inverse Kinematics problem for any type of serial robot manipulators was developed using a method based on nonlinear programming technique and forward recursion formulas, singular configurations does not affect the solution generated by this method [12]. An iterative algorithm has been introduced on the basis of the global Newton's method to find staircase-like inverse solutions for a simple robots [13]. Offset modification method presented to solve inverse Kinematics problem for serial robot manipulator with any DOFs, it was shown that this method is suitable for real time manipulator control [14]. Genetic algorithm widely used for determining inverse position, velocity and acceleration [15], [16], [17] and [1]. The combination of using genetic algorithm and neural network also used for inverse positioning of robots manipulators, GA and ANN complete each other to find the final inverse position [18] and [19]. Kinematic analysis for a 5DOF robot using Multi-Layered Feed Forward Neural Network was studied [20]. A heuristic solution was presented for the inverse Kinematics problem, the heuristic consist on combination of the distance between the actual and the desired position of the gripper the best manipulability direction [21].

An algorithm for the closed loop inverse Kinematics problem has been presented, the study proposed an iterative method to determine Euler solution then taking the results to achieve the numerical integration by using Crank-Nicolson method to get better convergence for the differential inverse problem [22]. This paper deals with the issue of calculating the inverse acceleration problem for serial robot manipulator of any degree of freedom using the evolutionary algorithm harmony search.

3. HARMONY SEARCH ALGORITHM (HSA)

Harmony search (HS) is a jazz-mimicking algorithm that can be considered a special case of the genetic algorithm in which a suggested random proposed solution improved continuously until reach to the proper or optimal one. This process is similar to the musicians action when they try different combinations of music pitches stored in the mind [23] .also HS can be referred as an evolutionary algorithm that is its mechanism depends on evolution of an initial guess of solution to an optimal one. For many years (HS) was applied widely in different fields like image enhancement [24], optimization of the mobile robot [25], pipe network design [26] and many others.

4. HSA FOR INVERSE ACCELERATION METHOD

The purpose of this study is to optimize forward acceleration equation for a robot manipulators to return set of joint angular acceleration for the manipulator. The general forward acceleration equation for serial robot manipulators [1]:

$$\ddot{x} = J\ddot{q} + \dot{J}\dot{q} \text{ ……..(1)}$$

Where $\ddot{x}$ is a vector represents the acceleration components in the task space , J and $\dot{J}$ is the Jacobian and time derivative of the Jacobian respectively, joints angular acceleration and velocity expressed as $\ddot{q} = [\ddot{q}_1\, \ddot{q}_2\, \ddot{q}_3 \ldots \ddot{q}_n]T$ and $\dot{q} = [\dot{q}_1\, \dot{q}_2\, \dot{q}_3 \ldots \dot{q}_n]T$ respectively. It is known that:

$J = f(q)$ and $\dot{J} = f(q, \dot{q})$ otherwise :

$$\dot{q} = \int \ddot{q}\, dt \text{ ……..(2)}$$

$$q = \iint \ddot{q} dt \text{ ……...(3)}$$

So, q and $\dot{q}$ are functions of the acceleration in the joint space , hence acceleration vector in the task space is clearly as a function of robot specifications, which they are constants, and $\ddot{q}$ which is the only variable and take the form of

$$\ddot{q} = [\ddot{q}_1, \ddot{q}_2, \ddot{q}_3, \ldots . \ddot{q}_n]$$

In (HSA) an initial set of $\ddot{q}$ changed continuously until they meet the desired acceleration of the end-effector. The basic HS can be described as follows

1- Definition a matrix hold the minimum and maximum value for each joint acceleration in the robot

$$PVB = \begin{bmatrix} \ddot{q}_1^{min} & \ddot{q}_1^{max} \\ \ddot{q}_2^{min} & \ddot{q}_2^{max} \\ & \vdots \\ \ddot{q}_n^{min} & \ddot{q}_n^{max} \end{bmatrix} \text{…..(4)}$$

Where PVB is range of variables matrix, min is the minimum variable value, max is the maximum variable value and n is the DOFs of the manipulator. For each element minimum value chosen to be -1 and maximum value is 1.

2- initializing harmony matrix having (N,n) dimensions, this matrix is the memory of the algorithm

$$HM = \begin{bmatrix} \ddot{q}_1^1 & \ddot{q}_2^1 & \ldots \ddot{q}_n^1 \\ \ddot{q}_1^2 & \ddot{q}_2^2 & \cdots \ddot{q}_n^2 \\ \vdots & & \\ \ddot{q}_1^N & \ddot{q}_2^N & \ldots \ddot{q}_n^N \end{bmatrix} \text{……..(5)}$$

Where HM is harmony memory and N is number of rows (solutions) in the memoty and taken to be 50 and n equal to the DOFs of the robot manipulator. Each variable in HM take a random value between maximum and minimum limits for the corresponding variable in range of variables matrix

3- calculate fitness value for each row(solution) in the harmony memory by using fitness function.

4- new solution was generated based on harmony memory consideration rate HMCR and Pitching Adjust Rate PAR [5]

$$PAR = \frac{(PAR_{max} - PAR_{min})}{MaxItr * currentItr + PAR_{min}} \text{ ……(6)}$$

Where PAR_{max} is maximum pitching adjust rate and taken equal to 0.95, PAR_{min} is minimum pitching adjust rate and taken equal to 0.3, MaxItr is maximum iteration and equal to 104,and currentItr is the current number of iteration in the search loop.

5- Updating harmony memory with the newly generated solution , it will replace the worse element in HM if it has less fitness value otherwise the solution will eliminated.

6- repeat steps 4 to 5 until maximum iteration is reached.

5. EXAMPLE

A 3R planar robot manipulator was used as a case study to verify the ability of the proposed algorithm to handle an acceptable solution. Forward Kinematics equations for the robot was derived to positioning the end-effector in joint space where the robot controller operates. Figure (1) shows the frame assignment for RRR planar manipulator.

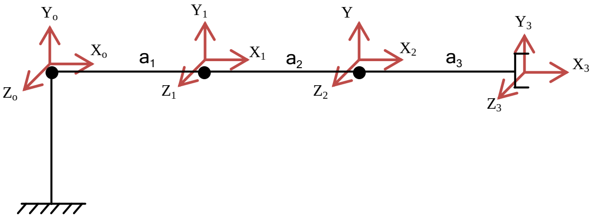


Figure (1) RRR frame assignment

Table (1) refer to the ng Denavit-Hartonberg parameters.

Table (1) Denavit-Hartonberg parameters

Link *i*	*αi*	*ai*	*di*	*θi*
O₁-O₀	0	a₁	0	θ_1
O₂-O₁	0	a₂	0	θ_2
O ₃-O₂	0	a₃	0	θ_3

Taken links length a1 = a2 = a3 = 1m.

By substitution these parameters in Denavit matrix, the transformation matrix for each link calculated, and serial multiplication of these matrices lead to the End-effector transformation matrix T_3^0 which is should be

$$T_3^0 = \begin{bmatrix} C_{123}, & -S_{123}, & 0, & a_1C_1 + a_2C_{12} + a_3C_{123} \\ S_{123}, & C_{123}, & 0, & a_1S_1 + a_2S_{12} + a_3S_{123} \\ 0\ , & 0\ \ , & 1, & 0 \\ 0\ , & 0\ \ , & 0, & 1 \end{bmatrix} \text{….(7)}$$

Where C, S refer to sin and cos and subscript 1, 12, 123 refer to q1, q1+2and q1+2+3 respectively , q is the generalized coordinate joint variable.

5.1 Equations Of Acceleration

It is clear from the above mentioned homogenous transformation matrix equation (7) that:

$$x=a_1C_1+a_2C_{12}+a_3C_{123} \text{........(8)}$$

$$y=a_1S_1+a_2S_{12}+a_3S_{123} \text{................(9)}$$

Equations (8) and (9) which represent tool tip motion in x-y plane will used to drive the components of the forward acceleration equation which is

$$\begin{bmatrix} \ddot{X} \\ \ddot{Y} \end{bmatrix} = J\ddot{q} + \dot{J}\dot{q} \text{ ....(10)}$$

Where $\ddot{X}$ and $\ddot{Y}$ are acceleration components in x and y directions. $\ddot{q}$ and $\dot{q}$ are column vectors represent angular acceleration and velocity respectively. The joints angular velocities and displacements can be evaluated from a given set of joint angular acceleration using this equations:

$$\dot{q}_1 = \ddot{q}_1 * t \text{ ....(11)}$$

$$\dot{q}_2 = \ddot{q}_2 * t \text{ ....(12)}$$

$$\dot{q}_3 = \ddot{q}_3 * t \text{ ....(13)}$$

$$q_1 = \ddot{q}_1 * \frac{t^2}{2} \text{ ....(14)}$$

$$q_2 = \ddot{q}_2 * \frac{t^2}{2} \text{ ....(15)}$$

$$q_3 = \ddot{q}_3 * \frac{t^2}{2} \text{ ....(16)}$$

Where t is the time interval, J is the Jacobian matrix and take the form

$$J = \begin{bmatrix} \frac{\partial x}{\partial q_1} & \frac{\partial x}{\partial q_2} & \frac{\partial x}{\partial q_3} \\ \frac{\partial y}{\partial q_1} & \frac{\partial y}{\partial q_2} & \frac{\partial y}{\partial q_3} \end{bmatrix} \text{ ....(17)}$$

Where

$$\frac{\partial x}{\partial q_1} = -a_1S_1 - a_2S_{12} - a_3S_{123} \text{ ....(18)}$$

$$\frac{\partial x}{\partial q_2} = -a_2S_{12} - a_3S_{123} \text{ ....(19)}$$

$$\frac{\partial x}{\partial q_3} = -a_3S_{123} \text{ ....(20)}$$

$$\frac{\partial y}{\partial q_1} = a_1C_1 + a_2C_{12} + a_3C_{123} \text{ ....(21)}$$

$$\frac{\partial y}{\partial q_2} = a_2C_{12} + a_3C_{123} \text{ ....(22)}$$

$$\frac{\partial y}{\partial q_3} = a_3S_{123} \text{ ....(23)}$$

The differential of the Jacobian matrix is

$$\dot{J} = \begin{bmatrix} \dot{J}_{11} & \dot{J}_{12} & \dot{J}_{13} \\ \dot{J}_{21} & \dot{J}_{22} & \dot{J}_{23} \end{bmatrix} \text{ ....(24)}$$

Where

$$\dot{J}_{11} = -a_1\dot{q}_1C_1 - a_2\dot{q}_{12}C_{12} - a_3\dot{q}_{123}C_{123} \text{ ....(25)}$$

$$\dot{J}_{12} = -a_2\dot{q}_{12}C_{12} - a_3\dot{q}_{123}C_{123} \text{ ....(26)}$$

$$\dot{J}_{13} = -a_3\dot{q}_{123}C_{123} \text{ ....(27)}$$

$$\dot{J}_{21} = -a_1\dot{q}_1S_1 - a_2\dot{q}_{12}S_{12} - a_3\dot{q}_{123}S_{123} \text{ ....(28)}$$

$$\dot{J}_{22} = -a_2\dot{q}_{12}S_{12} - a_3\dot{q}_{123}S_{123} \text{ ....(29)}$$

$$\dot{J}_{23} = -a_3\dot{q}_{123}S_{123} \text{ ....(30)}$$

Where $\dot{q}_i$, $\dot{q}_{ijk}$is joint angular velocity, i= 1,2 and i,j,k=1,2,3 is summation of multiple angular velocities.

6. FITNESS FUNCTION

The fitness function that is desired to be minimized by harmony search algorithm is illustrated as follows:

Pseudo code :

```
For ( first point in the trajectory to final point in the trajectory)
If abs($a_{opt}$)i-abs($a_{real}$)i < 0
$fitness = \sum_{i=1}^{n} a_{opt} - a_{real}$
End
  End
```

Where n number of points on the trajectory and a_opt is a vector represents acceleration values generated using equations from (10) to (30). while a_real is the given trajectory that should be transformed to the joint space and i is the current point in the trajectory having n points, while Fitness is the fitness function which has to be minimized and abs is the absolute value.

Figure (2) shows how the initial set of angular acceleration values made settled from random generated values to optimum ones while HS algorithm keep iterated.

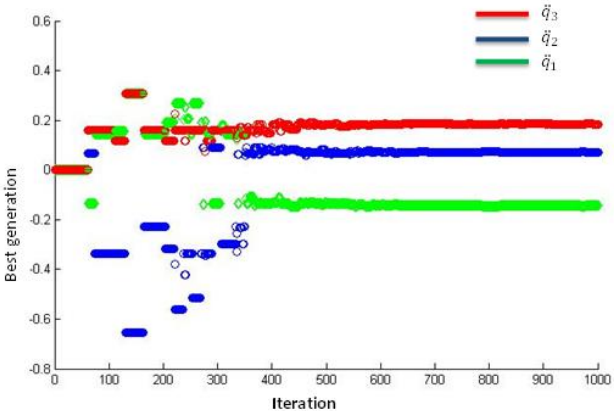


Figure (2) Best generations over sequence of iterations

7. RESULTS AND DISCUSSION

In order to examine the efficiency of the harmony search algorithm for predicting the inverse acceleration Kinematics many tests with different trajectories as well as simulation in hard conditions including complex trajectories and variation in geometrical parameters were done using models built using MATLAB. Due to multi-degree of freedom for the robot arm the multi-solution phenomenon was encountered but the optimized joints acceleration still producing a semi-matched trajectory with that given one, this should be enough for this paper because the original purpose is to estimate joints acceleration that produce the same or close acceleration trajectory to the desired one. Four trajectories was generated by a simulation MATLAB program using forward equations

(10) to (30) with time 0 up to 5 seconds by suggestion $\ddot{q} = [0.1, -0.15, -0.12]$, $\ddot{q} = [0.2, 0.6, -0.3]$, $\ddot{q} = [0.4, -0.7, -0.5]$, this yielded the trajectories shown in figures (3), (4) and (5).

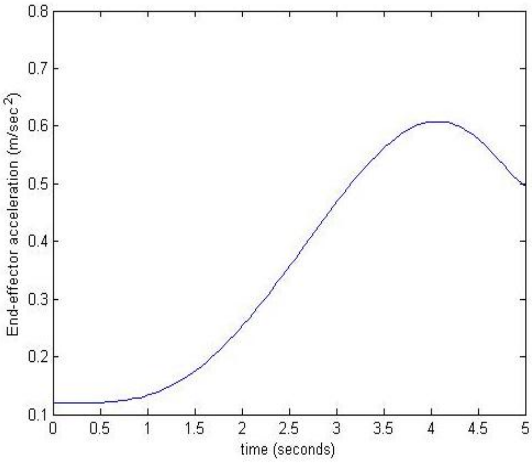


Figure (3) end effector acceleration for case one

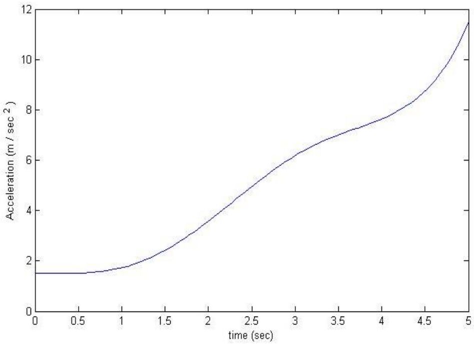


Figure (4) end effector acceleration for case two

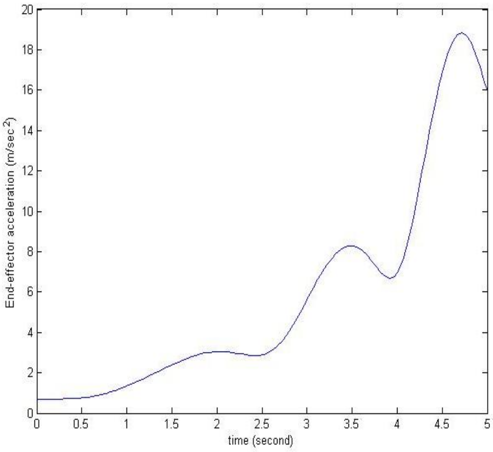


Figure (5) end effector acceleration for case

also changing the geometrical parameters which are links lengths to be a1=0.9 m, a2=0.7 m and a3=0.3 m and using $\ddot{q} = [-0.8, 0.1, -0.9]$ to generate the fourth acceleration trajectory shown in figure (6).

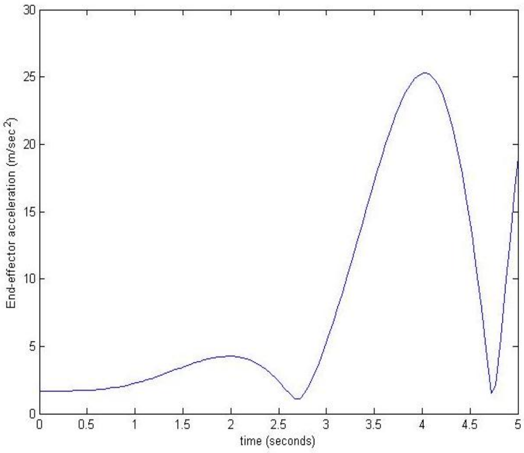


Figure (6) end effector acceleration for case four

All the four acceleration trajectories will be used as an input to the HSA for the inversion process.

Case one: The trajectory shown on figure (3) was used in the HS to calculate set of joints acceleration of the robot manipulator. it represents a_real parameter in the fitness function. Multiple solutions was estimated by harmony search algorithm one of them is $\ddot{q}_1 = 0.102$, $\ddot{q}_2 = -0.1585$ and $\ddot{q}_3 = -0.1087$. The corresponding trajectory for this specific solution is well matched that one in figure (3) as shown in figure (7).

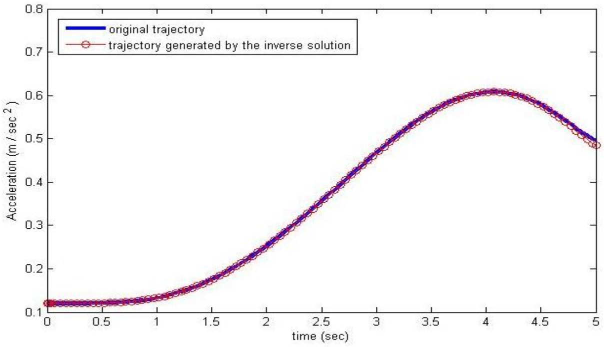


Figure (7) original and proposed trajectory for case one

Case two: The solution for the graph in figure (4) was found that it is $\ddot{q}_1 = 0.7999$, $\ddot{q}_2 = -0.3$ and $\ddot{q}_3 = -0.2999$. the trajectory for this solution is well matched with that graph in figure (4) as illustrated in figure (8).

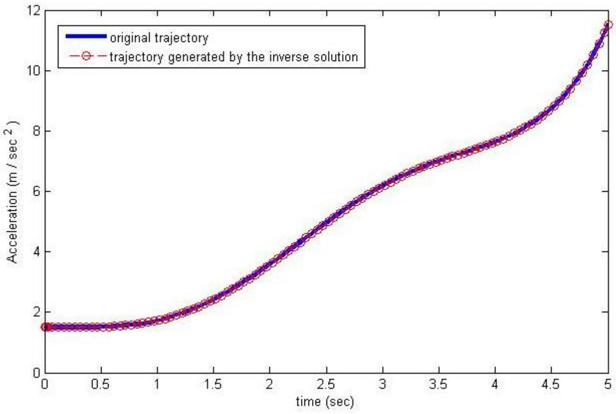


Figure (8) original and proposed trajectory for case two

Case three: Figure (5) is more complicated trajectory has been inverted by HSA, the obtained solution for this graph is $\ddot{q}_1 = -0.7999$, $\ddot{q}_2 = 0.5$ and $\ddot{q}_3 = 0.7$ which gives the optimized trajectory. The two trajectories shown in figure (9)

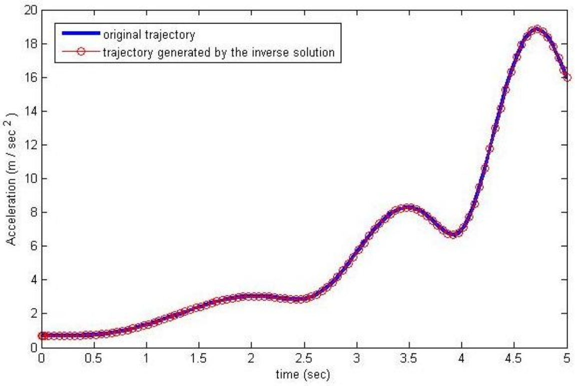


Figure (9) original and proposed trajectory for case

Case four: acceleration trajectory in figure (6) was inverted and give the solution $\ddot{q}_1 = -0.8026$, $\ddot{q}_2 = 0.1067$ and $\ddot{q}_{3=} - 0.9051$ the corresponding trajectory for this solution and the original one which is illustrated in figure (6) shown in figure (10).

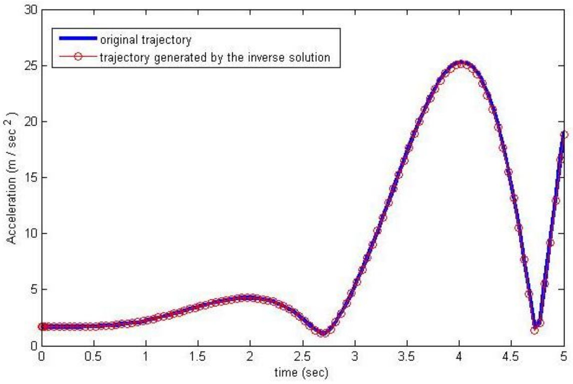


Figure (10) original and proposed trajectory for case four

Both desired trajectories and optimized ones by the (HS) for each case was tested in ANSYS 15.0 software package for further checking . figure (11) shows a 3D model for RRR robot manipulator in ANSYS environment.

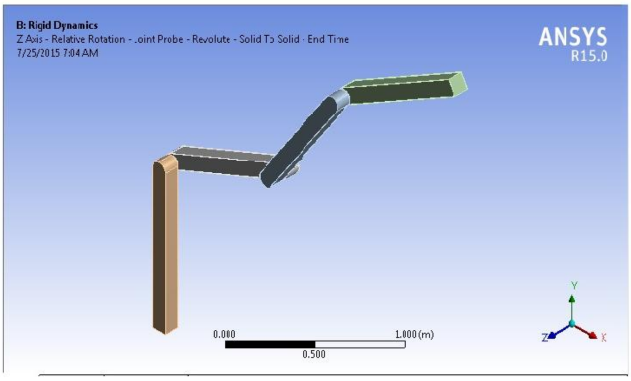
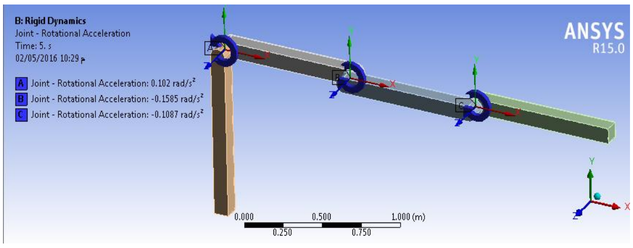

B: Rigid Dynamics
Z Axis - Relative Rotation - .oint Probe - Revolute - Solid To Solid - End Time
7/25/2015 7:04 AM

Figure (11) RRR robot manipulator model in ANSYS

Rigid dynamics solver in ANSYS support forward Kinematics so by setting joints loads which are in our case the rotational acceleration, the whole robot with its three links start moving for a specific time see figure (12).


B: Rigid Dynamics
Joint - Rotational Acceleration
Time: 5. s
02/05/2016 10:29 م

A Joint - Rotational Acceleration: 0.102 rad/s²
B Joint - Rotational Acceleration: -0.1585 rad/s²
C Joint - Rotational Acceleration: -0.1087 rad/s²

Figure (12) Adding rotational acceleration to robot joints

The four cases mentioned previously was simulated , the result is matched what is obtained from MATLAB program. Figure (13) shows the desired trajectory which is obtained by setting joint loads to $\ddot{q} = [0.1, -0.15, -0.12]$ and the optimized one which is obtained by setting joint loads to is $\ddot{q}_1 = 0.102$, $\ddot{q}_2 = -0.1585$ and $\ddot{q}_3 = -0.1087$. The two trajectories were matched what is obtained in figure (7).

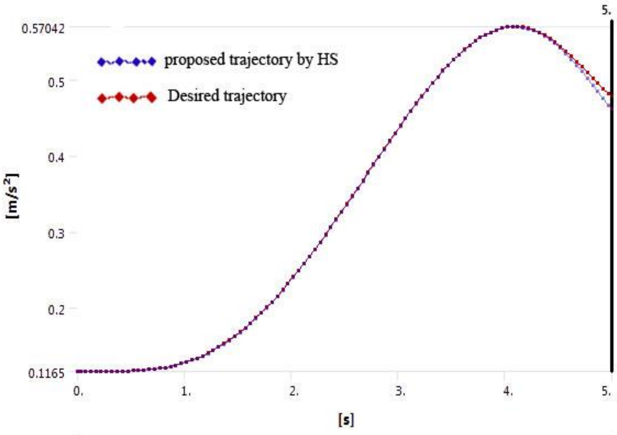


Figure (13) End-effector acceleration for case one simulated in ANSYS

The data generated for the above cases are program controlled with run time from 0 to 5 seconds separated to 103 points, all the points on each trajectory and the proposed one by the harmony search algorithm were examined and showed a great deal of matching as shown in tables (2) and (3) which represent error percentage for some chosen points on the trajectories for the four cases. For this specific problem i.e. inverse acceleration HSA appear to be an efficient way to solve the problem in addition this method explicitly include solving the other part of the problem of the inverse Kinematics for a given trajectory i.e. inverse position and velocity.

Table (2) Error percentage between points on the original trajectory and the proposed trajectory by the HSA for case one and two

Time Sec	Case one			Case two		
	Acceleration on the original trajectory m/sec^2	Acceleration on the proposed trajectory by the HSA m/sec^2	error %	Acceleration on the original trajectory m/sec^2	Acceleration on the proposed trajectory by the HSA m/sec^2	error %
0.12	0.1200	0.1197	0.2500	1.5000	1.4998	0.0133
0.37	0.1202	0.1199	0.2510	1.5045	1.5043	0.0134
0.87	0.1275	0.1271	0.2776	1.6341	1.6338	0.0157
1.37	0.1604	0.1598	0.3530	2.1872	2.1868	0.0215
1.87	0.2295	0.2286	0.4034	3.2482	3.2473	0.0255
2.37	0.3277	0.3264	0.389	4.5967	4.5955	0.0271
2.87	0.4390	0.4376	0.3121	5.8905	5.8889	0.0280
3.37	0.5404	0.5395	0.1737	6.8267	6.8247	0.0294
3.87	0.6018	0.6018	0.004	7.4558	7.4537	0.0293
4.37	0.5930	0.5927	0.0545	8.3647	8.3631	0.0191
4.87	0.5147	0.5046	1.9522	10.5189	10.5181	0.0067

Table (3) Error percentage between points on the original trajectory and the proposed trajectory by the HSA for case three and four

Time Sec	Case three			Case four		
	Acceleration on the original trajectory m/sec^2	Acceleration on the proposed trajectory by the HSA m/sec^2	error %	Acceleration on the original trajectory m/sec^2	Acceleration on the proposed trajectory by the HSA m/sec^2	error %
0.12	0.7002	0.6999	0.0428	3.1003	3.0998	0.0161
0.37	0.7180	0.7177	0.0410	3.1343	3.1339	0.0153
0.87	1.1175	1.1173	0.0203	4.0026	4.0025	0.0025
1.37	2.1112	2.1109	0.0117	6.6766	6.6769	0.0051
1.87	2.9752	2.9746	0.0179	9.6675	9.6675	0.0005
2.37	2.8626	2.8613	0.0473	10.1271	10.1394	0.1214
2.87	4.5698	4.5689	0.0203	13.2524	13.3300	0.5852
3.37	8.0936	8.0933	0.0033	33.1635	33.2014	0.1144
3.87	6.7472	6.7446	0.0374	54.1397	54.0023	0.2538
4.37	14.1555	14.1530	0.0172	48.2974	48.1799	0.2433
4.87	17.8225	17.8179	0.0257	44.0850	44.6630	1.3111

8. REFERENCES

[1] YONG-GUI ZHANG, YU-MEI HUANG, LI-MING XIE. “ROBOT INVERSE ACCELERATION SOLUTION BASED ON HYBRID GENETIC ALGORITHM”. Proceedings of the Seventh International Conference on Machine Learning and Cybernetics, Kunming. pp 2099-2013 July 2008.

[2] Panfeng Zhang, Xihui Mu, Zhenshu Ma, Fengpo Du. “An Adaptive PSO-Based Method for InverseKinematics Analysis of Serial Manipulator”, Quality, Reliability, Risk, Maintenance, and Safety Engineering (ICQR2MSE), 2012 International Conference on, pp. 122 – 1126, Chengdu, 15-18 June 2012.

[3] A. BAZERGHI, A. A. GOLDENBERG, and J. APKARIAN, ” An Exact Kinematic Model of PUMA 600 Manipulator”, IEEE TRANSACTIONS ON SYSTEMS, MAN, and CYBERNETICS, VOL. SMc-14, No.3, pp. 483-487MAY/JUNE 1984

[4] C.S.G. LEE, M. ZIEGLER, “Geometric approach in solving inverse Kinematics of PUMA robots”, IEEE TRANSACTIONS ON AEROSPACE AND ELECTRONIC SYSTEMS VOL. AES-20, NO. 6, pp.965-706, NOVEMBER 1984

[5] ANDREW A. GOLDENBERG, B. BENHABIB, and ROBERT G. FENTON, “A Complete Generalized Solution to the Inverse Kinematics of Robots”, IEEE JOURNAL OF ROBOTICS AND AUTOMATION, VOL. RA-1, NO. 1, pp. 14-20, MARCH 1985.

[6] VICTOR LOVASS-NAGY, SENIOR and R. J. SCHILLING, “Control of Kinematically Redundant Robots

Using (1)-Inverses", IEEE TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNETICS, VOL. SMC-17, NO. 4, p.p 644-649, JULY/AUGUST 1987.

[7] Pyung H. Chang, "A Closed-Form Solution for Inverse Kinematics of Robot Manipulators with Redundancy", IEEE JOURNAL OF ROBOTICS AND AUTOMATION, VOL. RA-3, NO. 5,p.p 393-403 OCTOBER 1987.

[8] VASSILIOS D. TOURASSIS and CHARLES P. NEUMAN, "Inverse Dynamics Applications of Discrete Robot Models", IEEE TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNETICS, VOL. SMC-15, NO. 6, pp. 798-803, NOVEMBER/DECEMBER 1985

[9] VlJAY KUMAR and JOHN F GARDNER, "Kinematics of Redundantly Actuated Closed Chains", IEEE TRANSACTIONS ON ROBOTICS AND AUTOMATION. VOL 6. NO 7 . pp. 269-274 APRIL 1990.

[10] CHI-HAUR WU and KUU-YOUNG YOUNG, "An Efficient Solution of Differential Inverse Kinematics Problem for Wrist-Partitioned Robots", IEEE TRANSACTIONS ON ROBOTICS AND AUTOMATION, VOL. 6, NO. 1, pp.117-123, FEBRUARY 1990.

[11] ZORAN R. NOVAKOVIC AND BOJAN NEMEC, "A Solution of the Inverse Kinematics Problem Using the Sliding Mode", IEEE TRANSACTIONS ON ROBOTICS AND AUTOMATION. VOL 6. NO 2, pp. 247-252, APRIL 1990.

[12] Li-Chun Tomy Wang and Chih Cheng Chen, "A Combined Optimization Method for Solving the Inverse Kinematics Problem of Mechanical Manipulators", IEEE TRANSACTIONS ON ROBOTICS AND AUTOMATION, VOL. I , NO. 4, pp. 489-499, AUGUST 1991

[13] Toyosaku Isobe, Kengo Nagasaka, and Shinji Yainamoto, "A New Approach to Kinematic Control of Simple Manipulators", EEL TRANSACIIONS Oh SYSTEMS, MAN, AND LYBEKNbIICS, VOL 22, NO 5, pp. 1116-1124, SEPTEMBEWOCTOBER 1992.

[14] Gregory Z. GrudiE and Peter D. Lawrence, "Iterative Inverse Kinematics witb Manipulator Configuration Control", IEEE TRANSACTIONS ON ROBOTICS AND AUTOMATION, VOL. 9, NO. 4, pp. 476-483, AUGUST 1993.

[15] Kevan A. Buckley, Simon H. Hopkins and Brian C. H. Turton, "Solution Of Inverse Kinematics Problems Of A Highly Kinematically Redundant Manipulator Using Genetic Algorithms", Genetic Algorithms in Engineering Systems: Innovations and Applications, 2-4 September 1997, Conference Publication No. 446, pp. 264-269, IEEE, 1997.

[16] F. Chapelle, P. Bidaud, "A Closed Form for Inverse Kinematics Approximation of General 6R Manipulators using Genetic Programming", Proceedings of the 2001 IEEE International Conference on Robotics & Automation Seoul, Korea. pp. 3364-3369, May 21-26, 2001.

[17] YONG-GUI ZHANG, W-ME1 HUANG, YI-ZHONG LIN, XIANG CHENG, FENG GAO, "AN APPROACH FOR ROBOT INVERSE VELOCITY SOLUTION USING GENETIC ALGORITHM", Proceedings of the Third International Conference on Machine Learning and Cybernetics, pp. 2944-2948, Shanghai, 26-29 August 2004.

[18] P.Kalra and Neelam Rup Prakash, "A Neuro-genetic Algorithm Approach for Solving the Inverse Kinematics of Robotic Manipulators", Systems, Man and Cybernetics, 2003. IEEE International Conference on Volume:2, pp. 1979 – 1984, IEEE, 5-8 Oct. 2003.

[19] Marija Tomic, Branko Miloradovic and Marija Jankovic, "Connectionist-Genetic Based Algorithm for Positioning Industrial Manipulator", 11th Symposium on Neural Network application in electrical engineering, NEUREL, pp. 59-64, September 2012.

[20] Rahul R Kumar, Praneel Chand, "Inverse Kinematics Solution for Trajectory Tracking using Artificial Neural Networks for SCORBOT ER-4u", Proceedings of the 6th International Conference on Automation, Robotics and Applications, pp.364-369, Feb 17-19, 2015, Queenstown, New Zealand.

[21] Claudiu Radu Pozna, Ernő Horváth, János Hollósi, "The inverse kinematics problem, a heuristical approach", SAMI 2016 • IEEE 14th International Symposium on Applied Machine Intelligence and Informatics, pp. 299-304, January 21-23, 2016 • Herl'any, Slovakia.

[22] Dániel András Drexler," Solution of the closed-loop inverse kinematics algorithm using the Crank-Nicolson method", SAMI 2016 • IEEE 14th International Symposium on Applied Machine Intelligence and Informatics, pp. 351-356, January 21-23, 2016 • Herl'any, Slovakia

[23] Z. W. Geem, J. H. Kim, and G. V. Loganathan, "A new heuristic optimization algorithm: Harmony Search," Simulation, vol. 76, no.2, pp. 60-68, 2001.

[24] Zaid Abdi Alkareem Y.A., Ibrahim Venkat, Mohammed Azmi Al-Betar and Ahamad Tajudin Khader, "Edge preserving image enhancement via Harmony", IEEE. 2012 4th Conference on Data Mining and Optimization 02-04 Sep.2012, Langkawi, Malaysia.pp 47-51.

[25] He Xu, Zhenyu Zhang and Dawei Tan, Xiaozhi Gao, Gaoliang Peng and Shuanghe Yu. "Optimization of Mobile Robot Based on Projection Method and Harmony Search". Proceedings of the 2008 IEEEInternational Conference on Robotics and Biomimetics, Bangkok, Thailand, pp 1653-1658. February 2009.

[26] Z. W. Geem, J. H. Kim, and G. V. Loganathan, "Harmony search optimization: application to pipe network design," International Journal of Modeling and Simulation, vol. 22, no. 2, pp. 125-133, 2002.