ENERGY OPTIMIZATION OF VISION GUIDED MANIPULATOR FOR OPTIMAL DYNAMIC PERFORMANCE

MUHAMMAD HERMAN JAMALUDDIN
MOHAMED AZMI SAID
MARIZAN SULAIMAN

Paper Presented at The Inaugural International Conference On The Roles Of The Humanities and Sosial Sciences in Engineering 2008, 5-6th December 2008, The Legend Hotel, Malaysia

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

Energy Optimization of Vision Guided Manipulator for Optimal Dynamic Performance

Muhammad Herman Jamaluddin
Kolej Universiti Teknikal Kebangsaan Malaysia
herman@kutkm.edu.my

Mohamed Azmi Said
Kolej Universiti Teknikal Kebangsaan Malaysia
azmisaid@kutkm.edu.my

Marizan Sulaiman
Kolej Universiti Teknikal Kebangsaan Malaysia
marizan@kutkm.edu.my

Abstract

This paper presents a step on how to optimize the energy and performance of an industrial robot. The project consist of three major phases; (1) theoretical, simulation and practical of forward and inverse kinematics for Fanuc LR Mate 200iB robot to determine their D-H parameters, (2) optimization of robot movements, and (3) implementation of practical tasks. The optimization process involves the control of two parameters known as position of joint angle (θ) and speed of the motor (ω) of the three (3) main axes of the robot. The energy is measured regarding 3 types of categorized movements known as reference, fixed and optimized which do the same repetitive task. Its forward and inverse kinematics problems will be simulated using Robotic tools of Matlab 7.0 and Roboguide V2.3.2. The simulation results will be compared with the real practical movements.

Keywords

Manipulators, Optimal Control, Path-Finding, Dynamic Performance.

Introduction

In this project, one of the sensors that are currently being used in controlling robot mechanism is vision. Optimal-based control strategies will be developed to perform manipulative actions. This vision guided robot could perform the tasks with improved efficiency using the optimal-based control strategies. The advantages of these optimal control strategies can be seen during performing the tasks with fewer movements of angles, better joint's motor speed and improved time motion that result in more efficient usage of energy.

To find the optimal dynamic performance of a robot, it is important to know the architecture of its kinematics first. From forward or inverse kinematics analysis of that particular robot, we will obtain an exact position and orientation for each of its movements. This will help to find the most suitable position and orientation to achieve optimal performance.

The outline of the paper is as follows. In section 2 a typical step on searching process of the manipulator's inverse kinematics with given end point is explained in detail. Simulation process is described in section 3 followed by energy measurement in section 4 and experimental result in section 5. Conclusions are stated in section 6.

Inverse Kinematics

Inverse Kinematics does the reverse of forward kinematics. Given the end point of the structure, what angles do the joints need to achieve that end point? It can be difficult, and there are usually many or infinitely many solutions. Most of the real systems are under constrained, so for a given goal position, there could be infinite solutions (i.e. many different joint configurations could lead to the same endpoint). The field of robotics has developed many inverse kinematics systems which, due to their constraints, have closed-form solutions.

One of the techniques that had been used to find the exact angle of each joint to reach the goal target is by using the Jacobian method which described briefly in [1].

Inverse Kinematics Analysis of Fanuc

After considering the Jacobian method trough the theoretical calculation based, the analysis which is using the Matlab Robotics tools were done. To find the inverse kinematics in Matlab, the function that had been used is `q = ikine(robot, T)`. This function returns the joint coordinates for the manipulator described by the object *robot* whose end-effectors homogeneous transform is given by T.

In this project, the robot needs to pick an object at position of [600, 0, -320] and place the object at position of [0, 620, -300].

The complete lists of function regarding this analysis of picking and placing the object using Matlab are as such:

```
%Entering the D-H parameter information for FANUC

L1=link([pi/2 150 0 0 0])
L2=link([0 250 0 0 0])
L3=link([pi/2 75 0 0 0])
L4=link([pi/2 0 0 290 0])
L5=link([4.7124 0 0 0 0])
L6=link([0 0 0 205 0])

%Searching for FANUC Position and Orientation Matrix
fanuc = robot({L1 L2 L3 L4 L5 L6})

%Generate the transform corresponding to a particular joint
coordinate
q=[0 0 0 0 0 0]

%Forward Kinematic
T=fkine(fanuc,q)

Td=transl(-35,0,615)*T


%Pick Position
T1=transl(160,0,-440)*Td

%Inverse Kinematic
q1=ikine(fanuc,T1)

%Place Position
T2=transl(-600,620,20)*T1
%Inverse Kinematic
q2=ikine(fanuc,T2)
```

The outputs for that executed function are as below:

```
L1 =
  1.570796   150.000000 0.000000  0.000000 R    (std)
L2 =

  0.000000   250.000000 0.000000  0.000000 R    (std)
L3 =

  1.570796   75.000000  0.000000  0.000000 R    (std)
L4 =

  1.570796   0.000000   0.000000 290.0000  R    (std)
L5 =

  4.712400   0.000000   0.000000  0.00000  R    (std)
L6 =
  0.000000   0.000000  0.000000 205.000000 R    (std)

fanuc =
noname (6 axis, RRRRRR)
         grav = [0.00 0.00 9.81]      standard          D&H
parameters

  alpha        A          theta          D       R/P
1.570796 150.000000    0.000000 0.000000 R    (std)
0.000000 250.000000    0.000000 0.000000 R    (std)
1.570796 75.000000     0.000000 0.000000 R    (std)
1.570796 0.000000      0.000000 290.0000 R    (std)
4.712400 0.000000      0.000000 0.000000 R    (std)
0.000000 0.000000      0.000000 205.0000 R    (std)

q =
     0      0      0      0      0      0

T =
    1.0000         0         0  475.0000
         0   -1.0000    0.0000    0.0023
         0   -0.0000   -1.0000 -495.0000
         0         0         0    1.0000

Td =
    1.0000         0         0  440.0000
         0   -1.0000    0.0000    0.0023
         0   -0.0000   -1.0000  120.0000
         0         0         0    1.0000

T1 =
    1.0000         0         0  600.0000
         0   -1.0000    0.0000    0.0023
         0   -0.0000   -1.0000 -320.0000
         0         0         0    1.0000

qi =

   -0.0000    0.3736    0.1848   -0.0000    0.5583   -0.0000

T2 =

    1.0000         0         0         0
         0   -1.0000    0.0000  620.0023
         0   -0.0000   -1.0000 -300.0000
         0         0         0    1.0000


q2 =

  -4.7124    5.5204   -3.9401   -0.0000    1.5802    1.5708
```

The output result above had shown that to go to the target position (pick) at coordinate [600,0,-320] from [440,0,120], the changes of joint angle as stated in *qi* are as such:

$\theta_1 = 0°$ $\theta_2 = 21.4°$ $\theta_3 = 10.6°$
$\theta_4 = 0°$ $\theta_5 = 32°$ $\theta_6 = 0°$

It is also shown that to go to the target position (place) at coordinate [0,620,-300] from [600,0,-320], the changes of joint angle as stated in *q2* are as such:

$\theta_1 = 270°$ $\theta_2 = 316°$ $\theta_3 = 225°$
$\theta_4 = 0°$ $\theta_5 = 90.5°$ $\theta_6 = 90°$

Simulation

The data from the changing angle on the joint will be simulated using Robotic Tools in Matlab. This is to ensure every joint will produce accurate movement and the required end point.

Figure 1 describes the output of the simulation for the pick position of simulated Fanuc robot. It shown that the end point is located at coordinate [600,0,-320]. While Figure 2 shows the output of the simulation for the place position and the end point is located at coordinate [0,620,-300].

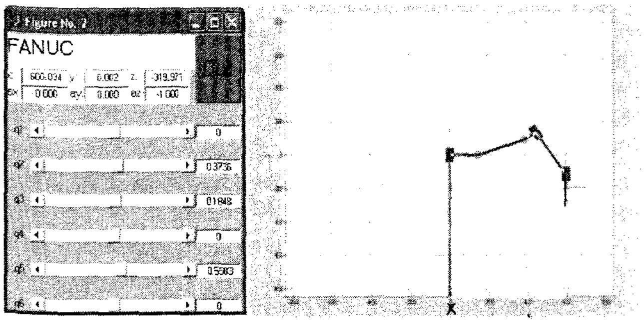


Figure 1 – Simulated movement for picking an object

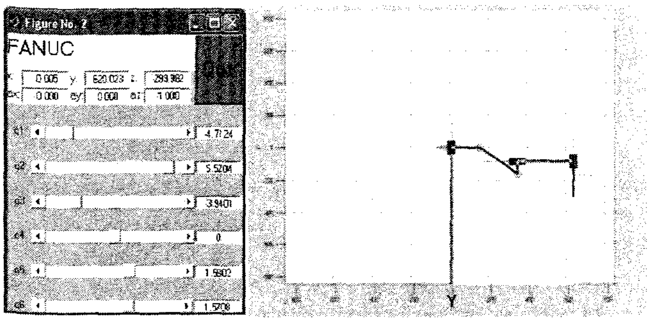


Figure 2 – Simulated movement for placing an object

From the data shown, the default position for the pick and place task is in Figure 1 and Figure 2. This default position is later described in text as Reference Movement. There are two types of movement that are modified so that it could achieve the same end point but with different joint angles. It is later described as Fixed Movement and Optimized Movement.

Energy Measurement

While the robot pick an object at position of [600, 0, -320] and place it at position of [0, 620, -300], the energy of overall task were measured using the Fluke 434 Three Phase Power

Quality Analyzer meter as shown in Figure 3. This analyzer meter has 4 BNC-inputs for current clamps and 5 banana-inputs for voltages.

The robot movements for pick and place application involving the use of many motor of its joint and including the energy of its controller that attached together with the robot. So, this measurement was done for total of energy usage by the robot and its controller. The energy was measured at the input cable of the 3-phase power supply. Figure 4 explained how the connections of Analyzer are made to the 3-phase distribution system and Figure 5 described the real connection of Analyzer to the supply of the robot.

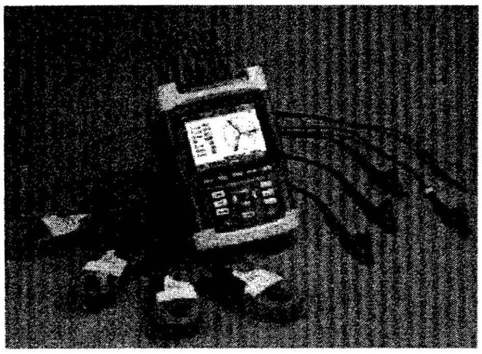


Figure 3 – Fluke 434 Three Phase Power Quality Analyzer

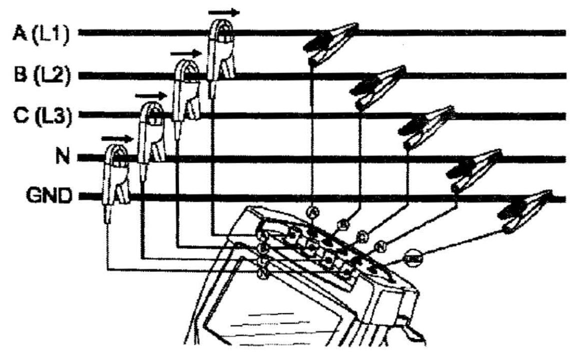
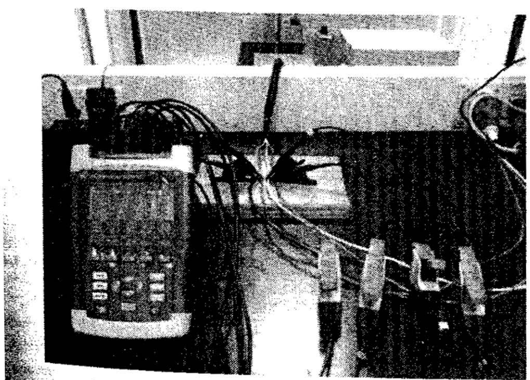


Figure 4 – Connection of Analyzer to 3-phase distribution system



ure 5 –*Real Connection* of Analyzer to 3-phase supply

erimental Result

n the data and information on simulation results d...

previously, it is time to test the movement of the manipulator in real application and measure their energy usage. The experiments were divided into three categories of movement which is Reference Movement, Fixed Movement and Optimized Movement. Those were experimented using real movement of Fanuc robot and run simultaneously together with Roboguide software. Each movement encompasses of seven steps from P[1] to P[7].

The steps are as such:

- P[1] – Home position
- P[2] – Pick object
- P[3] – Back to home position
- P[4] – Rotate 90° left
- P[5] – Place object
- P[6] – Home position of 90° left
- P[7] – Back to home position

Table 1 – End point coordinates (mm)

Step	X	Y	Z
P[1]	440	0	120
P[2]	600	0	-320
P[3]	440	0	120
P[4]	0	440	120
P[5]	0	620	-300
P[6]	0	440	120
P[7]	440	0	120

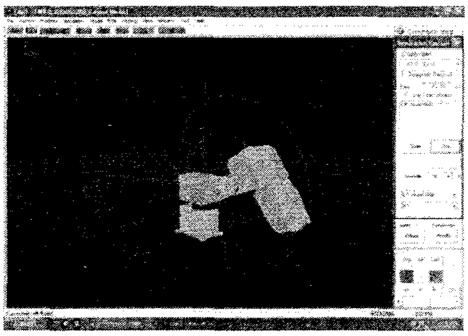
Table 1 shows the position of end point in coordinate format of [X,Y,Z] for each step. Those steps were considered as one cycle of task. While completing the task, the total energy usage by the system were measured and recorded. This measurement depends on process that involves the control of two parameters known as position of joint angle (θ) and speed of the motor (ω) of the 3 main axes of the robot. For each category of movement, the experiments are run with

- $\omega 1$=50% of motor's speed within one hour
- $\omega 2$=100% of motor's speed within 15 minutes.

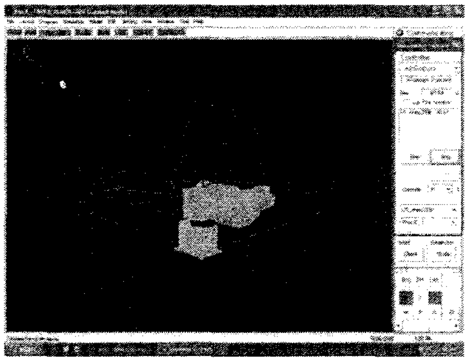
The experiment of picking and placing the object will be repeated until it reaches the time limit. Within that time frame, the task cycle will be counted and recorded to show the effectiveness of those movements.

Reference Movement

The first category (Reference Movement) was done following the seven steps above. Figure 6 describe the Reference Movement while picking (P[2]) and placing (P[5]) an object by Roboguide software. This Reference Movement exactly follows the movement of default inverse kinematics result.



(a)



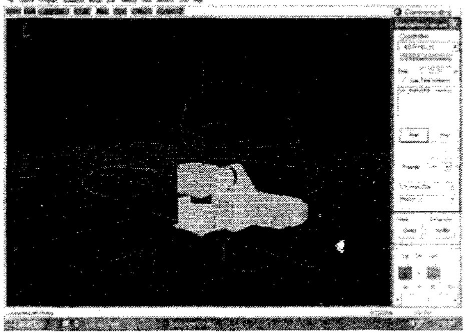
(b)

Figure 6 – References Movement for (a) picking object, (b) placing object

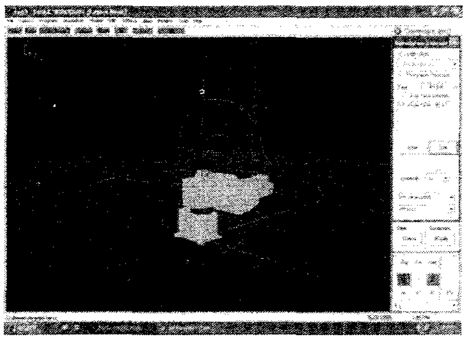
Fixed Movement

This type of movement was fixed to minimize the number of motor usage for the joint angle of 3 main axes to reach the target. So, it involves only the movements of motor for Joint 2.

It was done following the seven steps as stated earlier. Figure 7 describe the reference movement while picking (P[2]) and placing (P[5]) an object. There was a different kind of method on how the robot does the picking task that can be found from Figure 6 (a) and Figure 7 (a). However, the same method is used for the robot placing the object as shown in Figure 6 (b) and Figure 7 (b). Whatever it is, the robot can still find the same end point target for picking and placing the object.



(a)

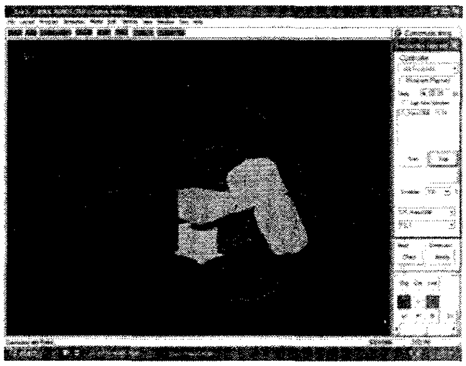


(b)

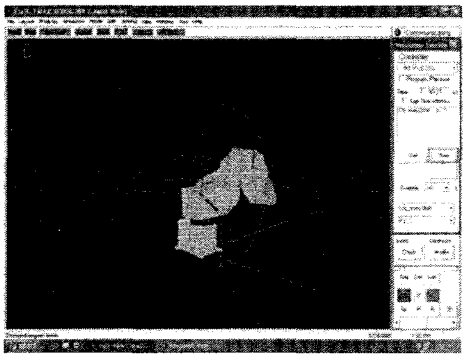
Figure 5 – Fixed Movement for (a) picking object, (b) placing object

Optimized Movement

To see the difference from the movement that had been done earlier, this movement were designed to minimize the changing of angle for each joint. Figure 7 shows the Optimized Movement for picking and placing an object.



(a)



(b)

Figure 7 - Optimized Movement for (a) picking object, (b) placing object

Performance Criteria

Those experiments were run whilst the number of cycle for completing an hour task were recorded with different speed of the motor. That means all type of the movement will be executed twice with different value of motor's speed. For this experiment, we had chosen the 50% and 100% of

Table 2 – Energy measurement for each movement category

Movement	Time	Motor Speed	Cycle	Energy Measured (kWh)	Energy Usage (kWh)	1 Cycle Energy Usage (kWh)	1 Cycle Time	Optimization Percentage
Idle	1 hr	-	-	0.147	-	-	-	-
Reference	1 hr	50%	135	0.192	0.045	0.000333	26.67s	0.00%
Fixed	1 hr	50%	109	0.193	0.046	0.000422	33.03s	-26.61%
Optimized	1 hr	50%	170	0.190	0.043	0.000253	21.17s	24.12%
Idle	15min	-	-	0.037	-	-	-	-
Reference	15min	100%	65	0.057	0.020	0.000312	13.85s	6.54%
Fixed	15min	100%	53	0.055	0.018	0.000344	16.98s	-3.30%
Optimized	15min	100%	81	0.055	0.018	0.000225	11.11s	32.41%

motor's speed.

The data on Table 2 shows the Energy measurement and time motion taken for completing the cycle and repeating it within specific hour with different speed of the joint's motor.

It is seen that both Optimized Movement of 50% and 100% of motor speed shows the less cycle time taken for completing the task which are 21.17s and 11.11s respectively. It also shown that within one hour of repetitive task, it can produce a number of 170 task cycle which more than the other two movements.

The Optimized Movement can run single task only in 11.11s and a number of 81 cycles within fifteen minutes which produce 32.41% of energy saving depend on the Reference Movement (50% of motor's speed). That's more efficient than Reference Movement and Fixed Movement. It can be stated here that the Optimized Movement can produce better energy optimization for the entire category.

Conclusion

This paper describes that to find the specific angle of FANUC robotic arm with given target coordinate, the inverse kinematics solution needs to be applied. The simulation technique and calculation method much helped to achieve the objective of this project. From the experiments that had been executed, it shows that the movement with less task's cycle time and with the fastest of joint motor's speed is more efficient in their overall dynamic performance. The lesser the motor's movement on a joint, especially on the three main axes, the lesser the time is taken to complete the task and save energy as well as increase the manipulator's performance. This performance can be really optimized by measuring the smallest energy usage for each of the category of movement.

Acknowledgments

The authors wish to acknowledge the financial assistance provided by Kolej Universiti Teknikal Kebangsaan Malaysia for the overall support of this research.

References

[1] Craig, J.J., 1989. Introduction to Robotics: Addison-Wesley

[2] Shiller and Sundar, April 1991. Design of Robotic Manipulators for Optimal Dynamic Performance. In Proceedings of the IEEE International Conference on Robotics and Automation, 344-349. Sacramento, California.

[3] Ngunyen and Graefe, November 2001. Object Manipulation by Learning Stereo Vision-Based Robots. In Proceedings of the IEEE International Conference on Intelligent Robots and Systems, 146-151. Maui, Hawaii, USA.

[4] Ngunyen and Graefe, April 2000. Self-Learning Vision-Guided Robots for Searching and Grasping Objects. In Proceedings of the IEEE International Conference on Robotics and Automation, 1633-1638. San Francisco.

[5] Bowling and Khatib, April 1997. Design of Macro / Mini Manipulators for Optimal Dynamic Performance. In Proceedings of the IEEE International Conference on Robotics and Automation, 449-454. Albuquerque, New Mexico.

[6] Herman, Azmi, Marizan and Horng, Jun 2006. Vision Guided Manipulator for Optimal Dynamic Performance. In Proceedings of the IEEE Student Conference on Research and Development, 147-151. Selangor, Malaysia.