Journal of Mechanical Research and Application (JMRA),Vol. 13 No.2, 1402(2023),1-5

Camera Attitude Control of a Flying 3R Robotic Manipulator

Armin Safaei
MS Graduate in Mechatronics
Department of Mechanical Engineering, South Tehran Branch, Islamic Azad University, Tehran, Iran

ArminSafaei@gmail.com

S. Mahdi Khorsandijou *
Assistant Professor
** Corresponding Author*
Department of Mechanical Engineering, South Tehran Branch, Islamic Azad University, Tehran, Iran
Email: m_khorsandijou@azad.ac.ir
Email: mahdi.khorsandijou@gmail.com

Ahmad Reza Khorshidvand
Associate Professor
Department of Mechanical Engineering, South Tehran Branch, Islamic Azad University, Tehran, Iran
Email: ar_khorshidvand@azad.ac.ir
Email: dr.khorshidvand@gmail.com

***Abstract*—Attitude of a camera attached to the end-effector of a 3R spatial robotic manipulator mounted on a flying platform with six DoF (Degrees-of-Freedom) has been controlled by NMPC (Nonlinear Model Predictive Control) algorithm. Motion of the closed-loop system has been numerically stimulated by MATLAB software. The obtained results show the success of this controller for the system. The controller has not been designed, but the design parameters have just been chosen to ensure numerical stability for the camera to track an earth-fixed object. The Lagrangian of the system has been derived by Maple software and the governing differential motion equations of the system have been found thereafter.**

Keywords*— Spatial Robotic Manipulator, Degrees of Freedom, Nonlinear Model Predictive Control (NMPC)

I. INTRUDUCTION

Lagrange-Euler, Newton-Euler and D'Alembert methods have been used for the dynamic formulation of manipulators. A Symbolic computational method has been presented by Filip [1] to obtain the dynamics equations of an n-DoF manipulator with revolute and prismatic joints in Mathematical software using Lagrange-Euler method. Driving join torque has been calculated for a certain imposed trajectory.

Strong nonlinearity of the dynamics equations sophisticates the control of the manipulators. Many algorithms have been suggested for the position, orientation and exerting or contact force control of these systems [2]. Poignet and Gautier [3] have proposed an approach in nonlinear model predictive control for a manipulator. Fateh and Izadbakhsh [4] have studied a robust control approach for state space equations of a high-speed manipulator. Qin and Badgwell [5] have presented an overview of commercially available model predictive control technology. The theoretical, computational, and implementational aspects of nonlinear model predictive control strategy has been reviewed by Allgöwer et. al. [6].

The present article discusses the attitude NMP control of a camera attached to the end-effector of a 3R spatial robotic manipulator mounted on a flying platform with six Degrees of freedom. The camera should be so oriented to take a picture from a fixed point object in the middle part of the picture framework. By the use of the Lagrange method, the governing differential motion equations of the system have been derived in Maple software. Matlab function "fminunc.m" has been used for optimization of NMPC to produce the required manipulator joint driving torques. The control parameters have been chosen as that of Matlab default values.

Only the obtaining of the numerical stability of the closed-loop system has been taken into consideration in this setting. Numerical stimulation of the closed-loop system has been provided by Matlab software.

II. PRINCIPLE OF MPC

MPC (Model Predictive Control) law is the exact or numerical solution of a finite receding horizon, open loop, optimal control problem satisfying the input and state constraints. It is subjected to the actuator, i.e. the required manipulator joint driving torques, of the system. The basic principle of the MPC has been depicted by Fig. 1. Based on the measurements obtained at time t, the controller predicts the system dynamics behavior in future over prediction horizon Tp and determines input over a control horizon Tc such that the predetermined open-loop performance is optimum **(**Tc ≤Tp).

If the MPC optimum law exists, then the input signal found at t = 0 can be applied to the open-loop system over t ≥0 when there is no disturbance and mismatch between model and plant. However, due to disturbance and model-plant mismatch the actual system behavior is different from what predicted. The joint driving torques, i.e. optimum open-loop input, is only exerted until the next sampling instant. The sampling time between each optimization might vary in principle. The optimization problem is re-evaluated after each sampling time, namely δ. By the use of the new system stated at time t+δ, the whole procedures, namely prediction and optimization, are repeated and the control and prediction horizons are shifted forward.

In Fig. 1 the open-loop optimal input is depicted as an arbitrary time function. The numerical solution for MPC law is usually parameterized by a finite number of base-functions. This is leaded to a finite-dimensional optimization problem. Piecewise constant input is often used in practice. This leads to decision for the value of Tc/δ In the problem. The determination of MPC law is on the basis of the predicted system behavior under some input and state constraints. This is a constrained finite-dimensional optimization problem, i.e. conditional desired cost-function minimization problem.

The finite prediction horizon is arbitrarily chosen, as a result, the system predicted behavior will generally differ from that of the closed-loop one. So, precaution must be taken into consideration to achieve closed-loop numerical stability and reasonable performance.

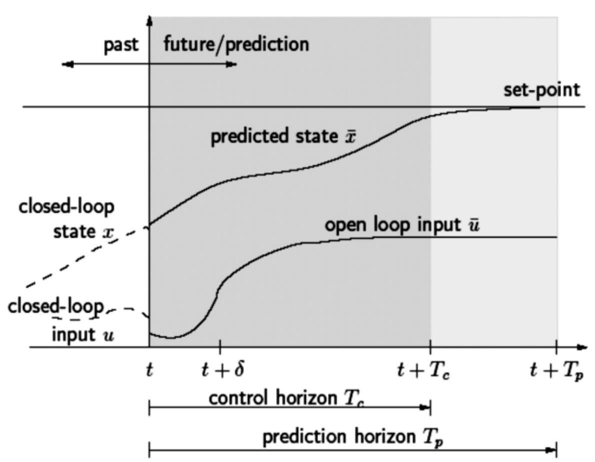

Fig. 1. Principle of model predictive control [8]

III. NMPC

A standard NMPC estimates the system states by a nonlinear model, and repeatedly calculates the optimum input, i.e. driving torque, that minimizes the desired cost-function over the prediction horizon. This input is implemented until the next sampling instant.

NMPC allows the direct use of nonlinear models for prediction. It allows the explicit consideration of input and state constraints. In NMPC specified time-domain-performance criteria is minimized in an on-line manner. The predicted behavior is generally different from that of the closed-loop system. Application of NMPC requires a real-time solution for the open-loop optimal control problem.

Many of these properties can be seen as advantages as well as drawbacks of NMPC. The possibility to directly use a nonlinear model is advantageous if a detailed first principles model is available. In this case often the performance of the closed loop can be increased significantly without much tuning. Nowadays first principle models of a plant are often derived even before a plant is build. Especially in the process industry there is a strong desire to use (rather) detailed models from the first design up to the operation of the plant for reasons of consistence and cost minimization. On the other side, if no first principle model is available, it is often impossible to obtain a good nonlinear model based on identification techniques. In this case it is better to fall back to other control strategies like linear MPC [8].

To perform the prediction the system states must be measured or estimated [٨].

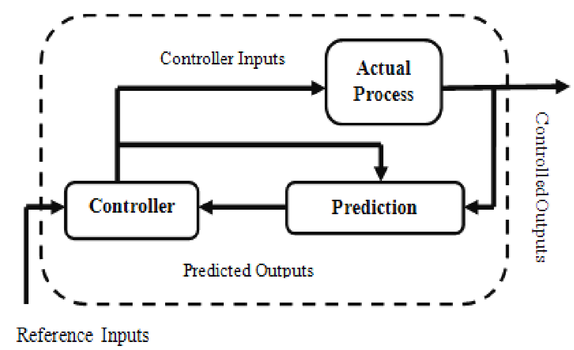

Fig. 2. MPC Block diagram [7]

IV. EQUATIONS OF 3R MANIPULATOR

Fig. 3 simply shows a 3R robot, Denavit-Hartenberg parameters for this robot have been derivate presented by table I.

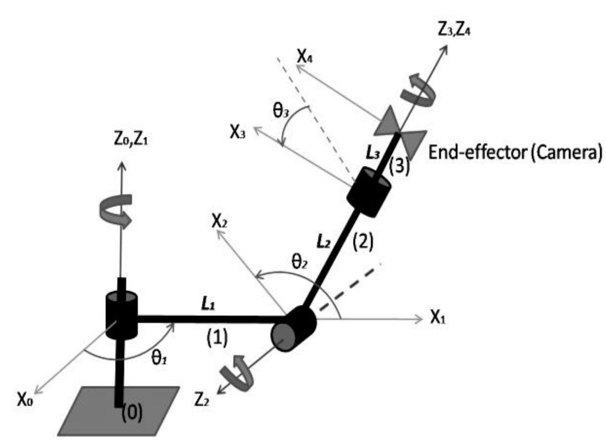

Fig. 3. Allocated frame and robot joint angles [7]

Table I. DERIVED DENAVIT-HARTENBERG PARAMETERS

	Link Parameters of spatial Manipulator			
I	$\boldsymbol{\alpha_{i-1}}$	$\boldsymbol{a_{i-1}}$	$\boldsymbol{d_i}$	$\boldsymbol{\theta_i}$
1	0	0	0	θ_1
2	90	l_1	0	θ_2
3	90	0	l_2	θ_3
4	0	0	l_3	0

It is intended to construct the transformation ${}^{0}_{4}T$ that defines frame {i} relative to the frame {i-1}. In general, this transformation will be a function of the four link parameters [9].

As it can be observed ${}^{E}_{0}T$ transform include of a rotation and a transmission. As result complete transform matrix which can give us an expression from end effectors to ground is calculated by:

$$ {}^{E}_{0}T\,{}^{0}_{4}T = {}^{E}_{4}T \qquad (1) $$

The Lagrangian dynamic formulation provides a means of deriving the equations of motion from a scalar function called the Lagrangian, which is defined as the difference between the kinetic and potential energy of a mechanical system. In our notation, the Lagrangian of a manipulator is:

$$ L(\Theta,\dot{\Theta}) = K(\Theta,\dot{\Theta}) - u(\Theta) \qquad (2) $$

Thereafter deriving robotic manipulator motion equation by MAPLE, we have used from MATLAB to solve the obtained equation. The outcome of MAPLE can be described by following expression:

$$ [A(\Theta)]_{9\times9}[\ddot{\Theta}]_{9\times1} = \left[B(\Theta,\dot{\Theta},\tau)\right]_{9\times1} \qquad (3) $$

Where matrix A indicate to the only position coefficients, $\Theta = (x, y, z, \alpha, \beta, \gamma, \theta_1, \theta_2, \theta_3)$, matrix B indicates to the position, velocity and acceleration variables, $\dot{\Theta} = (\dot{x}, \dot{y}, \dot{z}, \dot{\alpha}, \dot{\beta}, \dot{\gamma}, \dot{\theta}_1, \dot{\theta}_2, \dot{\theta}_3)$ and system torques $\tau = (\tau_1, \tau_2, \tau_3)$, and matrix $[\ddot{\Theta}] = [\ddot{X}\,\ddot{Y}\,\ddot{Z}\,\ddot{\alpha}\,\ddot{\beta}\,\ddot{\gamma}\,\ddot{\theta}_1\ddot{\theta}_2\ddot{\theta}_3]^T$ indicates to the system acceleration matrix. Following expression describe the extended form of (3) by using MAPLE software.

V. ERROR DEFINITION FOR CONTROLLER

Regarding to structure of robotic manipulator and by admitting that Z_4axis pass through center of end effector, it is sufficient to this axis cross the object. So it means an end effectors always see the object. In order to, Z_4 axis pass through the object following angles, α and β, must turn to zero.

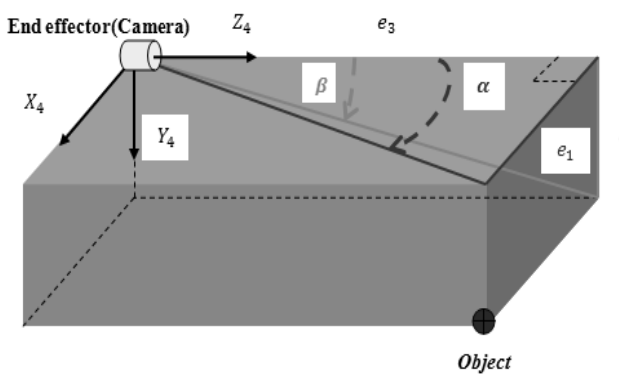

Fig. 4. Error angles [7]

Fig. 4 shows how an end effecter shall be moved by a control system in order to locate in object direction. Based on this Fig. 4 α, β shall became zero by controller to zero error in control system.

Regarding to pervious discussion to find the ${}_{object}^{4}T$, the achieved amount of ${}_{object}^{4}T(1,4)$, ${}_{object}^{4}T(2,4)$ and ${}_{object}^{4}T(3,4)$ are called e_1 , e_2 , e_3 and their derivations are called e_{1d} , e_{2d}, e_{3d}. Ideally in the control system we should have:

$$ \tan\alpha = \frac{e_1}{e_3} = 0 \qquad\qquad \tan\beta = \frac{e_2}{e_3} = 0 $$

In this experiment we call $\tan\alpha$ and $\tan\beta$, $error_1$ and $error_2$. If we zero amount of, $error_1$ and $error_2$, control will successfully apply to the system. According to Fig. 3 and by assuming that robotic manipulator actuator placed at robot joints, we need to accurate estimation for value of torque τ_2,τ_1. In is necessary to realize without precious value, successful robotic manipulator control would not happen.

Based on Fig. 3, to zero α, β angles and consecutively $error_1$ and $error_2$, τ_2 and τ_1 shall be predicted by controller.

VI. NUMERICAL SIMULATION

So far we have achieved the robotic manipulator motion equation and rest part of our project is to solve mentioned equation. In doing so, we used from scalar approach to maintain this purpose. But before that it is necessary to clarify some of the applied assumptions in this experiment.

The distance of object from a certain position would be known for designed control system and sudden movement of object and platform will be modeled as disturbance. Ideally, it is expected that the designed robot by processing of the available information from position of the object, track moving object so the allocated Z axis of camera cross the position of object continuously. In following equation we simply show that how we can find the distance of object from certain position when position of the object and point are available respect to a reference point.

$$[{}_{\text{object}}^{4}\mathrm{T}]_{4\times4} = [{}_{\mathrm{E}}^{4}\mathrm{T}]_{4\times4}[{}_{\text{object}}^{\mathrm{E}}\mathrm{T}]_{4\times4} \qquad (4)$$

In accordance pervious discussion, described by Fig. 3 and table. I, the estimated position matrixes $[{}_{\mathrm{E}}^{4}\mathrm{T}]_{4\times4}$ ، $[{}_{\text{object}}^{\mathrm{E}}\mathrm{T}]_{4\times4}$ can be simplify as follow:

$${}_{\text{object}}^{\mathrm{E}}\mathrm{T} = \begin{bmatrix} 1 & 0 & 0 & d_1 \\ 0 & 1 & 0 & d_2 \\ 0 & 0 & 1 & d_3 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

This expression shows that position of object respect to the earth only include transmission and d_1 ، d_2 and d_3 indicate to the position of object respect to the earth. Same thing happen for ${}_{\mathrm{E}}^{4}\mathrm{T}$.

VII. RESULTS

Velocity Variables								
$\dot{x}$	$\dot{y}$	$\dot{z}$	$\dot{\alpha}$	$\dot{\beta}$	$\dot{\gamma}$	$\dot{\theta}_1$	$\dot{\theta}_2$	$\dot{\theta}_3$
$1.5(^{m}/_{s})$	0	0	0	0	0	0	0	0

In this paper controlling of the robot manipulator has been simulated by substituting initial value. In this papers control variables, Tc (control horizon) and Tp (prediction horizon) have been assumed equal to 3. In doing so the amount 0.05 has been given to the (δ) value as sampling time and whole processing time (T) is equal to 7 seconds. Following snapshots have been taken from the Matlab which simply describe how the proposed control system operates. For an instance Fig. 5 and Fig. 7 are outputs of NMPC system which directly are applied to the controller. Meanwhile Fig. 6 and Fig. 8 are showing the controlled outputs of actual process which resulted by applying optimum control toques. For this simulation:

1. First it has assumed that the mass center of robot manipulator 1 and 2 are located at the beginning of links.

$$m_0 = a \quad , \quad m_1 = b \quad , \quad m_2 = c \quad , \quad m_3 = d$$

2. For each link regarding to first assumption, tensors assume as below.

$${}^{C_0}I_0 = J.\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}, {}^{C_1}I_1 = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}, {}^{C_2}I_2 = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}, {}^{C_3}I_3 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

3. The values of K (Spring Constant), Damping and g (gravity) are considered equal to zero.

4. The simulation related parameters values such as links mass, links length and also distance of object to the end-effector are presented in table II.

Table II. Values of M, L, J and Object distance

Mass (kg)				Lengths (m)			J	Object distance (m)		
M_0	M_1	M_2	M_3	L_1	L_2	L_3	J	d_1	d_2	d_3
20	1	1	1	0.25	0.25	0.25	2	300	400	-20

5. Table III contains the parameter values related to the robot spatial position into the earth.

Table III. Values of Positions

Position Variables								
θ_3	θ_2	θ_1	γ	β	α	z	y	x
0	0.17(R)	0.52(R)	0	0	0	500(m)	0	0

6. Table IV contains the parameter values related to the robotic manipulator velocity.

Table IV. Values of Velocities

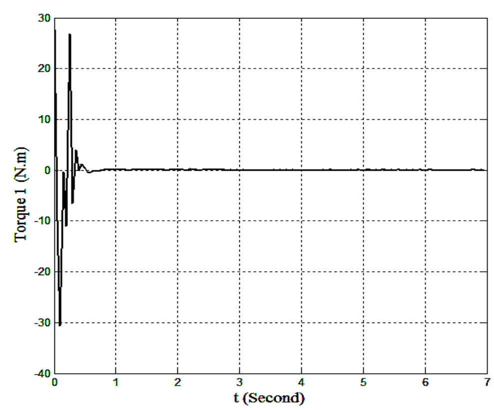

Fig. 5. First desired output of the Prediction model

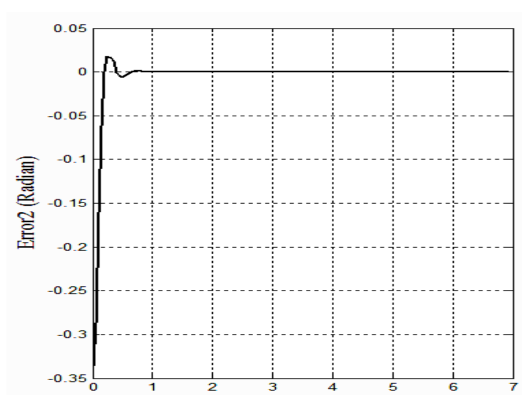

Fig. 6. Control Error 1

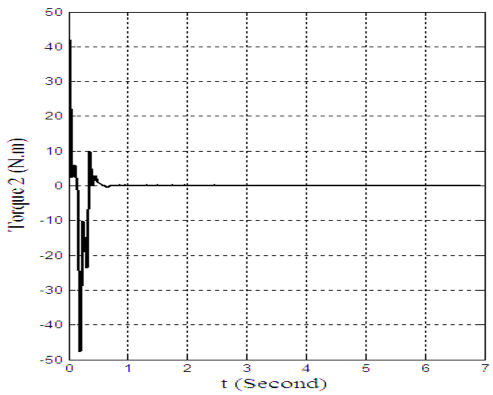

Fig. 7. Second desired output of the Prediction model

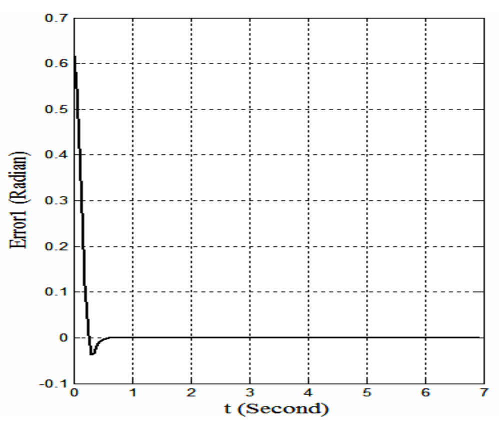

Fig. 8. Control Error 2

The suggested predictive model based on predefined target function, which are absolute value of errors and T_p=3, in each step produces optimized τ_1 , τ_2 for three stages. Since in NMPC controller, in each step, torques τ_1 ,τ_2 is applied as controller input to the system, this system by using optimized torques and Initial conditions calculates error_1 ,error_2. This process is repeated by (T/δ) until the desire outcome, which is zero, comes up. In every pair figure at above Fig. 5, Fig. 7 torques τ_1,τ_2 have been calculated and applied to the system. Results of these inputs are error_1 ,error_2 which have been presented by Fig. 6 and Fig. 8.

VIII. CONCLUTION

This study was conducted to control a camera which has been placed at the end of third link of 3R robotic manipulators as an end-effector. NMPC has been applied manipulator which has been mounted on a platform with six DoF to control robotic manipulators. This experiment was conducted when we have a moving platform which eventually leads to the movement on camera. Despite of moving object and platform in our experiment, achieved results show that the robot control system successfully directs the robot manipulator to track the object.

REFERENCES

[1] V. Filip, "Dynamic modeling of manipulators with symbolic computational mothod," Proceedings of the Romanian Academy, Series A, vol. 9, n. 3 , 2008.

[2] S. Ali A. Moosavian, Evangelos Papadopoulos, "Free-flying robots in space: an overview of dynamic modeling, planning and control," Robotica vol. 25, pp. 537-547, 2007.

[3] Ph. Poignet, M. Gautier, "Nonlinear Model predictive control of a robot manipulator," IEEE, Nagoya, Japan, pp. 401-406, April 2000.

[4] M. M. Fateh and A. Izadbakhsh, "Robust control of a high-speed manipulator in state space," World Academy of Science, Engineering and Technology, vol. 23, pp. 211-215, August 2007.

[5] S. Joe Qin and Thomas A. Badgwell, "An overview of industrial model predictive control technology," In Jeffery C. Kantor, Carlos E. Garcia, and Brice Caranham, editors, Fifth International Conference on Chemical Process Control, pp 232-256. AIChE and CACHE, 1997.

[6] F. Allgöwer, R. Findeisen, and Z. K. Nagy, "Nonlinear Model Predictive Control: From Theory to Application," J. Chin. Inst. Chem. Engrs., vol. 35, n. 3, 299-315, 2004.

[7] A. Safaei, "Camera Attitude Control With is Connected Six Dof Platform Using 3R Robot," M.Sc Dissertation, Islamic Azad University, South Tehran Branch, Mechatronics Group, February 2012.

[8] F. Allgower, R. Findeisen, and Z. K. Nagy, "Nonlinear model predictive control: From theory to application," JOURNAL-CHINESE INSTITUTE OF Chemical Engineers vol. 35, pp. 299-315, 2004.

[9] John J. Craig, "Introduction to Robotics, Mechanics and Control," Third Edition, Addison-Wesley, 2005.