Abstract: This work is focused on the motion control of an underactuated brachiation robot with 3 links. We present the modeling of the dynamics of the robot and introduce the application of the model-based predictive control (MPC) using a linearized model of the system. The robot has 3 revolute joints but only one of them is actuated, i.e., the robot has less control inputs than degrees of freedom. The investigation of the MPC to control such a system is due to the fact that MPC is able to deal with constraints (state or input limitations) in a direct way, obtaining an optimal control law. Also, it is investigated the use of a linearized model for prediction of system state. Simulation results are shown for complete arm switching, as well as some directions for future developments.

Keywords: brachiation robot, underactuated systems, MPC, linearization

1. INTRODUCTION

In the beginning of the past decade a new type of robot was proposed in (Fukuda et al. 1991b), presenting a six-link full-actuated brachiation robot that imitates the movement of an ape swinging from branch to branch. This new type of robot effectively used the gravity for swinging, instead of just compensate for it to reduce its influence. In the sequel a simpler two-link brachiation robot was proposed in (Fukuda et al. 1991a), where a control system using CMAC (Cerebellar Model Arithmetic Computer) was proposed, considering a heuristic creation of input driving forces in a previous training. Moreover, that two-link robot was designed as an underactuated system, i.e., the robot has two degree of freedom and only one control input, so it can move only dynamically. The control of underactuated mechanical systems is a research topic that during the past years has been in vogue in the literature, for example, (Oriolo and Nakamura 1991, Reyhanoglu et al. 1996, Su and Stepanenko 1999).

In (Saito et al. 1993) the feedback control also considered the direction of the arm to the target.

Another important work is (Nishimura and Funaki 1996, Nishimura and Funaki 1998), where an error learning method of the final-state control is applied to an underactuated robot. That paper considered a linearized state-space formulation with time-varying parameters. Those papers used a control strategy to track some reference trajectory that mimics a reference dynamical system. A fuzzy controller approach was presented in (Nakanishi et al. 1999). A more detailed analysis considering an underactuated brachiation robot is presented in (Odagaki et al. 1997).

The robot proposed here is intended to be used in the inspection of power transmission lines. There are several papers in literature developing robots for such task, but it is still open research topic. Similar work, with different solutions, are described in (Côté et al. 2000, Aracil et al. 2002, Campos et al. 2003, Rocha
and Siqueira 2004). The main advantage of the robot proposed in this paper is its capability to overcome obstacles on the supporting line due to its locomotion structure.

In this paper we apply the MPC scheme to an underactuated brachiation robot, with three-links: two arms and a body. The MPC scheme is adopted because it can deal with constraints and system limits directly during the control computation (Camacho and Bordons 1999) and generates an implicit optimal control law. Accordingly with the idea of take advantage of gravity forces to move the robot, it can be supposed that the optimal control would be one that makes effective use of such forces. Hence, the MPC would generate a control action that exploit the gravity forces.

It is known that nonlinear model predictive control (NMPC) is well developed, but the main drawback of this technique is that the computational effort necessary is much higher than the linear version. The NMPC deals with on-line nonlinear optimization, which is a nonconvex problem, with a large number of decision variables and a global minimum is in general impossible to find (Henson 1998). The strategy proposed in this work consists in using successive linearization approach, yielding a linear, time-varying description of the system to be solved by linear MPC.

This paper is organized as follows. In section 2 is developed the nonlinear dynamic model of the brachiation robot and the linearization technique using Taylor’s series expansion. In section 3 the model-based predictive control scheme is presented. In the sequel (section 4) some results of simulation are presented. Discussion concerning the results and the control scheme used and some conclusions are given in section 5.

2. UNDERACTUATED BRACHIATION ROBOT DYNAMICS

The 3-link underactuated brachiation robot considered in this work is shown in figure 1.

This robot has two arms and one link acting like a body where payload can be located. Each arm has a gripper that can attach firmly to the supporting line, allowing the robot to execute its movement in a way similar to a manipulator robot. It moves by releasing one arm from the supporting line and grasping it again in a forward point. It is important to keep in mind that although the robot has three joints, only one of them (joint 2) is actuated. In this section we present the nonlinear dynamic model for the robot and an estimated linearized model, which is based on Taylor expansion and then is discretized using Euler’s method.

2.1 Nonlinear Dynamic Model

The reference and coordinates systems of the robot is shown in figure 1. Considering the brachiation robot presented in this work can be written in a forward point. It is important to keep in mind that although the robot has three joints, only one of them (joint 2) is actuated. In this section we present the nonlinear dynamic model for the robot and an estimated linearized model, which is based on Taylor expansion and then is discretized using Euler’s method.

\[
D(\theta)\ddot{\theta}(t) + H(\theta, \dot{\theta}) + G(\theta) = \tau - F_v
\]  

(1)

Once we have an underactuated nonlinear system, it is suitable to define \( \tau \) such that \( \tau \rightarrow P \dot{u} \), with \( P = [0 1 1]^{T} \).

Thus, the dynamics of the robot, defined in (1), can be described in a general state space form as:

\[
\dot{q} = f(q, u)
\]  

(2)

where \( q = [\theta \ \dot{\theta}]^{T} \) is the state vector, \( u \) is the input of the system and

\[
f(q, u) = \left[ D^{-1}(\theta)(P \dot{u} - F_v - H(\theta, \dot{\theta}) - G(\theta)) \right]
\]  

(3)

The detailed development of the dynamics equations for the brachiation robot presented in this work can be seen in (de Oliveira and Lages 2006).

2.2 Linearized Dynamic Model

The linearized model around the point \((x_r, u_r)\) is obtained by expanding the right side of (2) in Taylor series and neglecting the high order terms, resulting in:

\[
\dot{x} = f(x_r, u_r) + \frac{\partial f(x, u)}{\partial x} \bigg|_{x=x_r, u=u_r} (x - x_r) + \frac{\partial f(x, u)}{\partial u} \bigg|_{x=x_r, u=u_r} (u - u_r)
\]  

(4)

By subtracting \( \dot{x}_r = f(x_r, u_r) \) from (4) we can write the linear model

\[
\dot{x} = f_x \dot{x} + f_u \dot{u}
\]  

(5)

where \( f_x \) and \( f_u \) are theJacobians of \( f(\cdot, \cdot) \) with respect to \( x \) and \( u \), respectively, evaluated around the point \((x_r, u_r)\), \( \dot{x} \equiv x - x_r \) and \( \dot{u} \equiv u - u_r \).
The approximation of $\dot{x}$ by using forward differences gives the following discrete-time system model:

$$\tilde{x}(k+1) = A(k)\tilde{x}(k) + B(k)\tilde{u}(k) \tag{6}$$

with

$$A(k) = T f_x(kT) + I \tag{7}$$

$$B(k) = T f_u(kT) \tag{8}$$

where $T$ is the sampling period and $k$ is the sampling interval.

2.3 On-line Parameter Estimation

One can observe that it is difficult to obtain the analytical expressions of the discrete model because it is necessary to compute the jacobian of $f(x,u)$ with respect to $x$ and $u$. Thus, we propose to estimate $A(k)$ and $B(k)$ using least square technique (Goodwin and Sin 1984). Furthermore, they depend on mass and inertia parameters of the robot, which are very difficult to determine exactly. Hence, the on-line estimation of $A(k)$ and $B(k)$ would also compensate for uncertain in the computation of $f(x_r,u_r)$. Figure 2 shows the proposed control scheme.

![Fig. 2. Block diagram of the proposed linear MPC controller.](image)

In order to proceed the estimation it is convenient to stack the elements of $A(k)$ and $B(k)$ and also to stack $\tilde{x}(k)$ and $\tilde{u}(k)$:

$$\Theta_i(k) = \begin{bmatrix} a_{i1}(k) \\ a_{i2}(k) \\ \vdots \\ a_{i6}(k) \\ b_i(k) \end{bmatrix} \quad \text{and} \quad \phi(k) = \begin{bmatrix} \tilde{x}_1(k) \\ \tilde{x}_2(k) \\ \vdots \\ \tilde{x}_6(k) \\ \tilde{u}(k) \end{bmatrix} \tag{9}$$

Hence, from (6) each state can be written as

$$\tilde{x}_i(k+1) = \phi^T(k)\Theta_i(k)$$

where $\Theta_i(k)$ is the vector of the unknown parameters and $\phi(k)$ is the linear regression vector.

Since the parameters $\Theta_i(k)$ are unknown, we want to obtain the estimates $\hat{\Theta}_i(k)$ by minimizing the cost

$$J_i(k,\Theta_i) = \sum_{j=0}^{k-1} \rho^{k-j} e_i^2(j) \tag{10}$$

where $e_i(j)$ is the estimation error given by

$$e_i(k) = \tilde{x}_i(k+1) - \phi^T(k)\hat{\Theta}_i(k) \tag{11}$$

and $\rho$ is the forgetting factor using to weight the importance of recent errors with respect to older errors. Typically, $0.9 < \rho \leq 1.0$.

Hence, the estimation problem can be formally stated as

$$\hat{\Theta}_i(k) = \arg\min_{\Theta_i} J_i(k,\Theta_i) \tag{12}$$

The solution of (12) is given in recursive form by (Goodwin and Sin 1984)

$$\hat{\Theta}_i(k+1) = \hat{\Theta}_i(k) + K(k) \left( \tilde{x}_i(k+1) - \phi^T(k)\hat{\Theta}_i(k) \right) \tag{13}$$

$$K(k) = \frac{P(k)\phi(k)}{\rho + \phi^T(k)P(k)\phi(k)} \tag{14}$$

$$P(k+1) = (I - K(k)\phi^T(k))P(k) \tag{15}$$

where $K(k)$ is the gain and $P(k)$ is the estimation error covariance. Once $\hat{\Theta}_i(k)$ has been estimated, $A(k)$ and $B(k)$ can be constructed from (9).

3. MPC CONTROL SCHEME

Model predictive control is an optimal control strategy that uses the model of the system to obtain an optimal control sequence by minimizing an objective function. At each sampling interval, the model is used to predict the behavior of the system over a prediction horizon. Based on these predictions, an objective function is minimized with respect to the future sequence of inputs, thus requiring the solution of a constrained optimization problem for each sampling interval.

Although prediction and optimization are performed over a future horizon, only the values of the inputs for the current sampling interval are used and the same procedure is repeated at the next sampling time. This mechanism is known as moving or receding horizon strategy, in reference to the way in which the time window shifts forward from one sampling time to the next one.

In this section we present the equations of the model-based predictive control (MPC) and then the application to the control of an underactuated brachiation robot.

3.1 MPC Problem Formulation

The basic elements present in all model-based predictive controller are: prediction model, objective
function and optimization procedure. The prediction model is the central part of the MPC, because it is important to predict the future outputs of the system. In this scheme, the state space model is used as prediction model, but in different MPC schemes, other models could be used (Camacho and Bordons 1999). The objective function defines the criteria to be optimized in order to force the generation of a control sequence that drives the system as desired.

Consider a general discrete nonlinear model, expressed as:

\[ x(k + 1) = f(x(k), u(k)) \quad (16) \]

where \( x(k) \) is the state vector and \( u(k) \) is the control input vector.

The objective function to be minimized assumes, in general, the following form:

\[
\Phi(k) = \sum_{j=1}^{N} x^T(k + j|k)Qx(k + j|k) \\
+ \sum_{j=1}^{N} u^T(k + j - 1|k)Ru(k + j - 1|k) \quad (17)
\]

where \( N \) is the prediction and control horizon and \( Q \geq 0 \) and \( R \geq 0 \) are weighting matrices that penalize the state error and the control effort, respectively.

By considering the fact that every real system is in practice subject to some constraint (for example physical limits), we define the following general constraint expressions:

\[
x(k + j|k) \in \mathcal{X}, \quad j \in [1, N] \\
u(k + j|k) \in \mathcal{U}, \quad j \in [0, N]
\]

where \( \mathcal{X} \) is the set of all possible values for \( x \) and \( \mathcal{U} \) is the set for all possible values for \( u \). By supposing that such constraints are linear with respect to \( x \) and \( u \), we can write:

\[
C_x x(k + j|k) \leq c, \quad j \in [1, N] \quad (18) \\
D_u u(k + j|k) \leq d, \quad j \in [0, N] \quad (19)
\]

Thus, the optimization problem, to be solved at each sampling time \( k \), is to find a control sequence \( u^* \) and a state sequence \( x^* \) such that minimize the objective function \( \Phi(k) \) under imposed constraints, that is:

\[
u^*_x, x^*_k = \arg \min_{u, x} \{ \Phi(k) \} \quad (20)
\]

subjected to:

\[
x(k|k) = x_0 \quad (21) \\
x(k + j|k) = f(x(k + j - 1|k), u(k + j - 1|k)), \quad j \in [1, N] \quad (22) \\
Cx(k + j|k) \leq c, \quad j \in [1, N] \quad (23) \\
Du(k + j|k) \leq d, \quad j \in [0, N] \quad (24)
\]

where \( x_0 \) is the value of \( x \) in instant \( k \).

The problem of minimizing (20) is solved for each sampling time, resulting in the optimal control sequence:

\[
u^* = \{ u^*(k|k), u^*(k+1|k), \ldots, u^*(k+N|k) \} \quad (25)
\]

and the optimal state sequence is given by:

\[
x^* = \{ x^*(k+1|k), \ldots, x^*(k+N|k) \} \quad (26)
\]

with an optimal cost \( \Phi^*(k) \). Thus, the control law defined by MPC is implicitly given by the first term of the optimal control sequence:

\[
h(\delta) = u^*(k|k) \quad (27)
\]

where \( h(\delta) \) is continuous during the sampling interval \( T \).

### 3.2 Brachiation Robot Control Using Linearization

The objective function takes into account the cartesian position of the end-effector of the robot, instead of considering directly the joint coordinates, because the robot must reach the supporting line \( (y = 0) \), independently of the joint configuration. It is important to highlight the fact the robot is not fully actuated. Thus, the objective function is given by:

\[
\Phi(k) = \sum_{j=1}^{N} X^T(k + j|k)QX(k + j|k) \\
+ \sum_{j=1}^{N} u^T(k + j - 1|k)Ru(k + j - 1|k) \quad (28)
\]

where \( X = [x\ y]^T \) is the cartesian coordinates vector, \( Q \) is a \( 2 \times 2 \) matrix and \( R \) is a real scalar.

The MPC deals with the system constraints as described by (18) and (19). Besides the fact that the system is underactuated, the control input has upper and lower bounds expressed by

\[
-\tau_{max} \leq \tau \leq \tau_{max} \quad (29)
\]

with \( \tau_{max} = 50Nm \). Moreover, we have restricted the joint angle to be within the interval:

\[
-\theta_{max} \leq \theta_i \leq \theta_{max} \quad (30)
\]

where \( \theta_{max} = \frac{\pi}{4} \) is the maximum angular displacement of each joint.

Rewriting the above equations according to (18) and (19) we obtain, respectively:

\[
D = \begin{bmatrix} 1 \\ -1 \end{bmatrix}, \quad d = \begin{bmatrix} \tau_{max} \\ \tau_{max} \end{bmatrix} \quad (31)
\]

and

\[
C = \begin{bmatrix} I \\ -I \end{bmatrix}, \quad c = \begin{bmatrix} \theta_{max} \\ \theta_{max} \end{bmatrix} \quad (32)
\]
4. SIMULATION RESULTS

The objective of this work is to investigate the use of MPC to control the motion of an underactuated brachiation robot. The robot moves switching the arms as a monkey does in the branch transfer motion. We initially posed the robot in the stretched position, with $\theta_1 = -\pi$, $\theta_2 = 0$ and $\theta_3 = 0$ and null velocities.

In this work we have implemented, using software Matlab$,^\circ$, the robot moving from the initial position to a fixed target position, such that the final position is in advance along $X$ axis and the end effector achieves the supporting line $y = 0$. The prediction horizon $N = 2$ is used, with a sample interval $T = 0.01s$. The gain matrix $Q$ used is set:

$$Q = \begin{bmatrix} 5 & 0 \\ 0 & 150 \end{bmatrix} \quad (33)$$

and $R = 1.0$. The gain matrix indicate we prioritize the end effector of the robot achieves the horizontal line, i.e., the cartesian position $y = 0$ instead of the robot advances in the $X$ coordinate.

One can observe in figure 3 that there are discontinuities in the angular coordinates, which occurs due to the switching of the arms. At the exact moment that the end effector reaches the horizontal line $y = 0$ closes the gripper. By changing the fixed hand of the robot, it is necessary to reassign the variable joints. Furthermore, it is important to see in figure 4 that when the arm switch happens, the angular velocities of joints 1 and 2 are zero (both hands are in contact with the horizontal line) and the last joint maintains the same angular velocity.

The position of the end effector of the robot in the $X$ and $Y$ axes as a function of time is depicted in figure 5 and the trajectory executed by the end effector in the XY-plan can be viewed in figure 6. The values of the objective function are plotted in figure 7. The arm switchings occur exactly at the minima values of the objective function. The unique torque applied to the system is on joint 2 and is shown in figure 8.

5. CONCLUSIONS AND FUTURE WORKS

This work considers the application Model-based Predictive Control (MPC) to motion control of an un-
deractuated brachiation robot with 3 links. According to the results of the previous section, it is possible to observe that the robot is able to release its hand from the supporting line and grasp it again in a forward point. It is shown that the robot is able to execute various cycles of motion, swinging and moving forward. Special attention must be taken to consider the gravitational force to reduce the external torque applied to the system, in order to maximize the time of autonomy of the robot. We intend to compare the computational effort of the linear and nonlinear model predictive control. A real-time implementation is also planned and a more detailed analysis concerning the constraints must be carried out.

6. ACKNOWLEDGEMENTS

The authors gratefully acknowledge the financial support from CAPES (Coordenação de Aperfeiçoamento de Pessoal de Nível Superior) and from CEEE (Companhia Estadual de Energia Elétrica).

REFERENCES


