Dynamics and Control of 2D SpiderCrane: A RHC Approach

Atul K. Kamath, Faruk Kazi, Navdeep M. Singh

Abstract— **In this paper we present modeling and control of a multicable suspended mechanism called the '2D Spider-Crane'. A spidercrane does not have any conventional heavy components and makes use of cables by virtue of which high transfer speeds are achievable. The initial part of the paper addresses the modeling of the mechanism, where the cable and pulley dynamics are separated, and the payload is viewed as a pendulum suspended from a cable whose suspension point lies on a mass that moves in a two-dimensional space. Using the Receding Horizon Control (RHC) strategy a trajectory tracking controller is proposed.Simulations were carried out in MATLAB.**

I. Introduction

The problem with the classical cranes is that the large inertia of the boom or the gantry limits rapid acceleration and deceleration since these may give rise to large inertial forces. For applications demanding fast weight handling, a new crane design has been proposed by the Laboratory of Automatic Control at *École Polytechnique Fédérale de Lausanne*, see [1]. Its main feature is the absence of heavy mobile components. The heavy elements of the mechanical structure are fixed and the positioning is done by cables that carry the load. As a result, this crane can work at considerably higher speeds. In this paper, we consider a planar version of the SpiderCrane consisting of two pylons as shown in Fig. 1. Henceforth, we refer to this mechanism as the 2D SpiderCrane.

The control and stability analysis of overhead cranes has recently received much attention. It is worth highlighting that this problem is not the same as that of other nonlinear mechanical systems like rigid robot manipulators. Cable cranes differ from rigid manipulators in a few significant ways

- Cable systems are usually underactuated.
- There is an additional set of dynamics - that of the pulleys.
- Cables can work only in tension and not apply compressive forces. (This requirement can be imposed as a holonomic constraint on the system).

A. K. Kamath is lecturer with Department of Electrical Engineering, Veermata Jijabai Technological Institute, Matunga, Mumbai,India. atulkamath84@gmail.com

Dr. Faruk Kazi is with Systems and Control Engineering, Indian Institute of Technology Bombay, Mumbai, India farukkazi@sc.iitb.ac.in

Dr. Navdeep M. Singh is Professor with the Department of Electrical Engineering, Veermata Jijabai Technological Institute, Matunga, Mumbai,India. nmsingh59@gmail.com

Motivated by the desire to achieve fast and precise payload positioning while minimizing swing, several researchers have proposed various control strategies for single cable, overhead crane systems, see [2], [3], [4]. The stabilization with swing minimization of a gantry crane is achieved using IDA-PBC methodology in [5], [6]. In this paper a coordinate change is proposed using partial feedback linearization to simplify the solution of partial differential equations arising out of matching conditions. The stabilization of a gantry crane system modeled with pulley dynamics leading to a holonomic constraint is addressed in [6]. A combination of IDA-PBC methodology and differential flatness based system inversion is employed in [7] for trajectory tracking of overhead gantry crane. A few other researchers have successfully proposed controllers synthesized from the differential flatness property of the systems to follow a specified trajectory, see [8], [9], [10]. A unified way for modeling a large class of cranes and weight handling mechanisms using Lagrange multipliers is presented in [10]. Differential flatness was originally introduced by Fliess *et al* [8] in a differentially algebraic context and later using Lie-Bäcklund transformations [11]. It represents a remarkable structural property of systems which are linearizable by means of endogenous feedback. Due to this property we can find a set of outputs such that we can express all states and inputs in terms of those outputs and their finite number of time derivatives. These outputs are called as flat or linearizing outputs and they are equal to the number of input channels. Flatness greatly facilitates the feedback controller design for a nonlinear multivariable system specifically in trajectory tracking control. Given some off-line planned trajectories for the flat outputs, it is possible to find a control law that achieves the objective of tracking the desired trajectories. A flatness-based approach to motion planning of VTOL aircraft is discussed in [12]. The differentially flat Lagrangian control systems with flat outputs that depend only on configuration variables are said to be configuration flat. A complete characterization of configuration flatness is provided in [13] for systems with n degrees of freedom and $n-1$ control inputs. In [14], [15] Sira-Ramirez *et al* elaborated on the possibilities of combining the flatness property of the system with other control methodologies like sliding mode control and passivity based control.

Most of the work reported in the literature focuses on the dynamics and control of single cable mechanisms. The first attempts at multiple cable mechanisms are found in [10] and [1]. In the attempt on the SpiderCrane at EPFL [1], a jet scheduling controller is proposed wherein a dynamical system called the jet-scheduler, provides the derivatives (the

ISBN 978-963-311-370-7

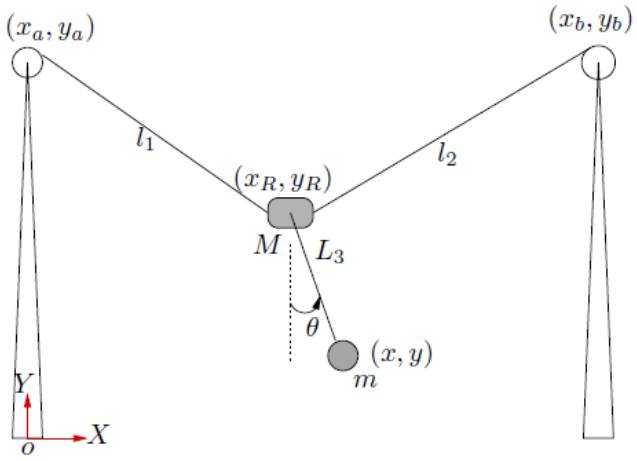


Fig. 1. 2D SpiderCrane mechanism

jets) of an ideal stabilizing flat-output trajectory instead of explicitly specifying the desired flat-output trajectory. Here we propose an alternate control scheme for such a multiple cable mechanism. The main contribution of this paper is to use RHC scheme to control the nonlinear dynamics of 2D SpiderCrane mechanism. As far as our knowledge goes, no one has applied and studied the RHC strategy on the 2D SpiderCrane problem.

The paper is organized as follows: Section II presents the dynamic model of the 2D SpiderCrane and formulates the problem. The payload and pulley dynamics are separated to arrive at a Euler-Lagrange formulation for the former. A brief overview of RHC Scheme is presented in Section III.Two control problems are addressed based on the notions of differential flatness, trajectory tracking and point-to-point transfer. Simulations and results are discussed in Section IV. Finally, we wrap up the report with some concluding remarks and future work in Section V.

II. 2D SpiderCrane Model

Consider a 2D SpiderCrane mechanism as shown in Fig. 1. The positioning of the load is done by adjusting the lengths l_1 and l_2. The model is underactuated and is subject to two holonomic constraints, and essentially captures all the control-theoretical perspectives of the SpiderCrane discussed in [16], [17].

Here, the position of the load is given by (x_P, y_P) with the load mass being m. The positions of the two motors are (x_a, y_a) and (x_b, y_b) with the corresponding rotary inertias taken as I_a and I_b. The ring has mass M and the position (x_R, y_R). The load is attached to the ring using a cable with fixed length of L_3. For the purpose of this study, we make the following assumptions:

1) The cable is massless and inelastic
2) Dissipative forces on the system are negligible
3) Both the pylons are assumed to be at the same height.

A. Dynamics of 2D SpiderCrane

We first develop the dynamic model for the complete system to bring out some intrinsic system dynamics related issues. We begin with the following set of configuration variables for complete system

$$q_C = [x_R \quad y_R \quad \theta \quad l_1 \quad l_2]^T \tag{1}$$

where $\theta \in [0, 2\pi)$ denotes the payload angle about the vertical axis, $x_R \in \mathbb{R}^1$ denotes the ring position along the X-coordinate axis, $y_R \in \mathbb{R}^1$ denotes the ring position along the Y-coordinate axis and l_1, l_2 represent the cable lengths. The control $u \in \mathbb{R}^2$ is defined as $u = [F_1 \quad F_2]^T$ where F_1 and F_2 represent the control-force inputs acting on the first and second cable, respectively. Note that the rotary actuation of the winch translates to the winding/unwinding action of the cable. The control objective is to move the payload from any position $q_{Ci} = [x_{Ri} \quad y_{Ri} \quad \theta_i \quad l_{1i} \quad l_{2i}]^T$ to the desired position specified as $q_{Cd} = q_{C*} = [x_{Rd} \quad y_{Rd} \quad \theta_d \quad l_{1d} \quad l_{2d}]^T$. At rest, the system has necessarily $\theta_d = 0$. The set of coordinates q_C as defined above is constrained by the following holonomic constraints:

$$\left.\begin{array}{rcl} C_1(q_C) & = & (x_R)^2 + (y_R - y_a)^2 - (l_1)^2 \quad = \quad 0 \\ C_2(q_C) & = & (x_R - x_b)^2 + (y_R - y_b)^2 - (l_2)^2 \quad = \quad 0. \end{array}\right\} \tag{2}$$

The Lagrangian of the system can now be expressed as

$$\mathcal{L}(q_C, \dot{q_C}) = \frac{1}{2}\dot{q_C}^T M(q_C)\dot{q_C} - V(q_C) \tag{3}$$

where

$$M(q_C) = \begin{bmatrix} M+m & 0 & mL_3\cos\theta & 0 & 0 \\ 0 & M+m & mL_3\sin\theta & 0 & 0 \\ mL_3\cos\theta & mL_3\sin\theta & {mL_3}^2 & 0 & 0 \\ 0 & 0 & 0 & I_a & 0 \\ 0 & 0 & 0 & 0 & I_b \end{bmatrix} \tag{4}$$

is the inertia matrix and the potential energy $V(q_C)$ is given as

$$V(q_C) = (M+m)gy_R - mgL_3\cos\theta. \tag{5}$$

The constraints (2) can be represented at the velocity level as $A^T(q_C)\dot{q}_C = 0$:

$$\begin{bmatrix} x_R & (y_R - y_a) & 0 & -l_1 & 0 \\ (x_R - x_b) & (y_R - y_b) & 0 & 0 & -l_2 \end{bmatrix} \begin{bmatrix} \dot{x_R} \\ \dot{y_R} \\ \dot{\theta} \\ \dot{l_1} \\ \dot{l_2} \end{bmatrix} = 0. \tag{6}$$

Note that $A(q_C)\lambda$ where $\lambda \in \mathbb{R}^2$ represents the constraint forces. The Lagrange multipliers $\lambda(t)$ are uniquely determined by the requirement that the constraints $A^T(q_C(t))\dot{q}_C(t) = 0$ have to be satisfied for all t. The annihilator of $A^T(q_)$ is generated by the span of the columns

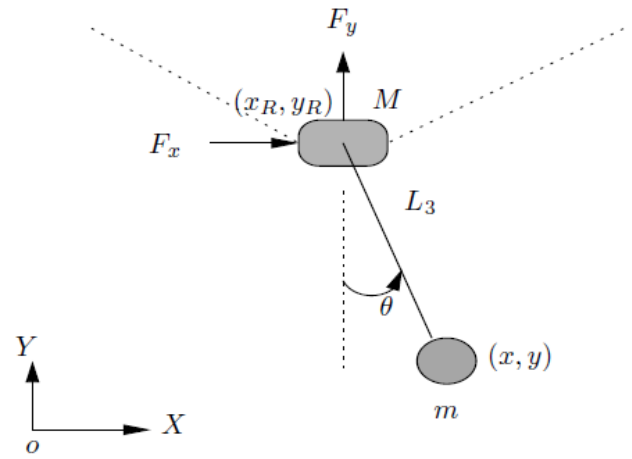


Fig. 2. 2D SpiderCrane gantry cart

of the matrix

$$S(q_C) \triangleq \begin{bmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & \frac{(y_R - y_a)}{l_1} & \frac{x_R}{l_1} \\ 0 & \frac{(y_R - y_b)}{l_2} & \frac{(x_R - x_b)}{l_2} \end{bmatrix}.$$

Hence, the admissible system motions lie in the range of $S(q_C)$ or, in other words, the vector $\dot{q_C}$ must be a linear combination of the columns of $S(q_C)$. Clearly for the following two configurations of the 2D SpiderCrane the Lagrange multipliers are not uniquely defined:

- the ring mass is at the first pulley with $l_1 = 0$
- the ring mass is at the second pulley with $l_2 = 0$.

Hence we exclude the above two configurations from our domain of operation.

B. Decoupled SpiderCrane Model

To facilitate designing a control law, we consider the 2D SpiderCrane as two subsystems as shown in Fig. 2 and Fig. 3. We first develop a control law considering F_x and F_y as control inputs. We pose our problem as the control of the payload mass m suspended by a cable from the ring of mass M on which we have the two actuating forces F_x and F_y. Henceforth, we shall refer to this subsystem as the *gantry mechanism*. Given the trajectories of F_x and F_y, we then compute the cable tensions using the geometry of the problem.

The configuration variables for the gantry mechanism are $q = (x_R, y_R, \theta)^T$ and the Lagrangian can be expressed as

$$\mathcal{L}(q, \dot{q}) = \frac{1}{2}\dot{q}^T M(q)\dot{q} - V(q) \tag{7}$$

where

$$M(q) = \begin{bmatrix} (M+m) & 0 & mL_3\cos\theta \\ 0 & (M+m) & mL_3\sin\theta \\ mL_3\cos\theta & mL_3\sin\theta & {mL_3}^2 \end{bmatrix}, \tag{8}$$

$$\text{and} \quad V(q) = (M+m)gy_R - mgL_3\cos\theta. \tag{9}$$

The resulting Euler-Lagrange equations are

$$\left.\begin{array}{rcl} F_x &=& (M+m)\ddot{x}_R + (mL_3\cos\theta)\ddot{\theta} \\ && -(mL_3\sin\theta)\dot{\theta}^2 \\ F_y &=& (M+m)\ddot{y}_R + (mL_3\sin\theta)\ddot{\theta} \\ && +(mL_3\cos\theta)\dot{\theta}^2 + (M+m)g \\ 0 &=& (mL_3\cos\theta)\ddot{x}_R + (mL_3\sin\theta)\ddot{y}_R \\ && +(mL_3{}^2)\ddot{\theta} + mgL_3\sin\theta. \end{array}\right\} \tag{10}$$

These Euler-Lagrange equations can be cast in the generic form for mechanical systems as

$$M(q)\ddot{q} + C(q, \dot{q})\dot{q} + \nabla V(q) = Gu. \tag{11}$$

where $M(q)$, $C(q, \dot{q})$, and $\nabla V(q)$, represent the inertia, centripetal-Coriolis, and potential energy terms, respectively. Note that the inertia matrix $M(q)$ is symmetric and positive definite.

We get a linear model of the 2D SpiderCrane model in the following manner.

$$\dot{x} = Ax + Bu \tag{12}$$

where,

$$A = \begin{pmatrix} 0 & I \\ -M(0)^1 \frac{\partial \nabla V(0)}{\partial \theta} & 0 \end{pmatrix} \tag{13}$$

$$B = \begin{pmatrix} 0 \\ M(0)^1 G \end{pmatrix} \tag{14}$$

C. Pulley Dynamics

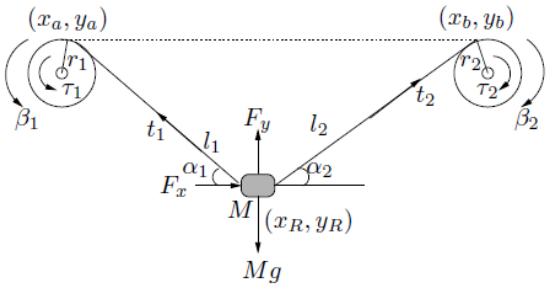


Fig. 3. Pulley-cable schematic

We now consider the dynamics of the second subsystem. Consider the cable and pulley mechanism as shown in Fig. 3. Here β_i and τ_i represent the pulley angle and motor torque exerted by the motor, respectively, for the i^{th} pulley. We can express the forces acting on the cart in terms of the cable tensions t_1 and t_2 as follows:

$$\begin{bmatrix} F_x \\ F_y \end{bmatrix} = \begin{bmatrix} -\cos\alpha_1 & \cos\alpha_2 \\ \sin\alpha_1 & \sin\alpha_2 \end{bmatrix} \begin{bmatrix} t_1 \\ t_2 \end{bmatrix} + \begin{bmatrix} 0 \\ Mg \end{bmatrix}, \tag{15}$$

where $\alpha_1, \alpha_2 (> 0)$ are as shown in Fig. 3. Our assumption that the α_is are greater than zero is based on practical considerations that the two cables during operation can never

be horizontal. With the rotary inertias I_a and I_b for the first and the second pulley, the pulley dynamics read

$$\begin{bmatrix} I_a & 0 \\ 0 & I_b \end{bmatrix} \begin{bmatrix} \ddot{\beta}_1 \\ \ddot{\beta}_2 \end{bmatrix} = \begin{bmatrix} \tau_1 \\ \tau_2 \end{bmatrix} - r \begin{bmatrix} t_1 \\ t_2 \end{bmatrix}. \quad (16)$$

Here $r_1 = r_2 = r$ is the pulley radius. Also, the no-slip constraint on the i^{th} pulley gives $r\dot{\beta}_i = \dot{l}_i$. Combining (15) and (16) with the no-slip constraint, we can express the motor torques in terms of F_x and F_y as follows:

$$\begin{aligned} \begin{bmatrix} \tau_1 \\ \tau_2 \end{bmatrix} &= r \begin{bmatrix} \cos\alpha_1 & -\cos\alpha_2 \\ -\sin\alpha_1 & -\sin\alpha_2 \end{bmatrix}^{-1} \begin{bmatrix} F_x \\ F_y - Mg \end{bmatrix} \\ &+ r \begin{bmatrix} I_a & 0 \\ 0 & I_b \end{bmatrix} \begin{bmatrix} \ddot{l}_1 \\ \ddot{l}_2 \end{bmatrix} \ (\alpha_i > 0, i = 1, 2). \quad (17) \end{aligned}$$

Note that $\ddot{l}_1$ and $\ddot{l}_2$ can be found from successive differentiation of the constraint equation given in (2).

We set the control objective of moving the payload from the initial position (x_{Pi}, y_{Pi}) to the desired position (x_{Pd}, y_{Pd}), while keeping the load swing angle as small as possible both during the transfer and after the desired position is achieved.

III. Receding Horizon-based Control Strategy

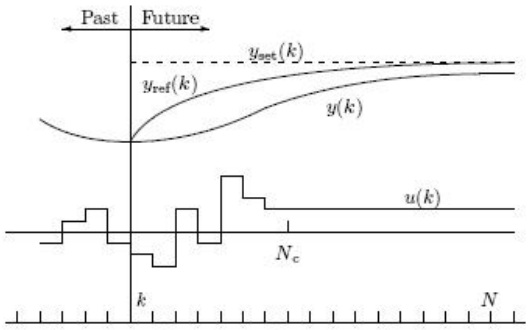
A. Introduction

Model predictive control is a control strategy developed around certain common key principles. First is explicit on-line use of a system model to predict the system output at future time instants. Secondly calculation of an optimal control action based on the minimization of one or more cost functions with or without constraints. The various MPC algorithms differ mainly in the type of model used to represent the system and its disturbances, as well as the cost functions to be minimized, with or without constraints.

An important difference between Model Predictive Control (MPC) and PID kind design method is the explicit use of a model. The more advanced control system of MPC can better track changes on the set point because it has knowledge on how the set point is going to change and how the system reacts to a given change in the control variable. When working with nonlinearities and multiple inputs and outputs this offers a set of smaller, easier-to-handle control problems that PID algorithms can address.MPC can provide high performance when models are accurate. Moreover, MPC is one of the very few strategies, which guarantee the accomplishment of constraint specifications

B. Basic operation

In general, the model predictive control problem is formulated as solving on-line a finite horizon open-loop optimal control problem subject to system dynamics and constraints involving states and controls. Figure 3 shows the basic principle of model predictive control. Based on measurements obtained at time t, the controller predicts the future dynamic behavior of the system over a prediction horizon N and determines (over a control horizon $Nc \leq N$) the input such that a predetermined open-loop performance objective functional is optimized. If there were no disturbances and no model-plant mismatch, and if the optimization problem could be solved for infinite horizons, then one could apply the input function found at time $t = 0$ to the system for all times $t > 0$. However, this is not possible in general. Due to disturbances and model-plant mismatch, the true system behavior is different from the predicted behavior. In order to incorporate some feedback mechanism, the open-loop manipulated input function obtained will be implemented only until the next measurement becomes available. The time difference between the recalculation/measurements can vary, however often it is assumed to be fixed, i.e the measurement will take place every δ sampling time-units. Using the new measurement at time $t + \delta$, the whole procedure prediction and optimization is repeated to find a new input function with the control and prediction horizons moving forward.



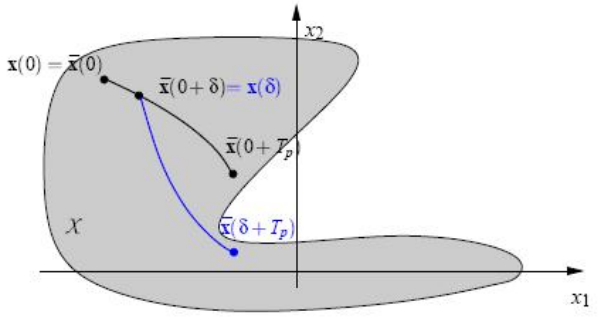
Fig(4): Principle of model predictive control.

The finite horizon open-loop optimal control problem described above is mathematically formulated as follows:

$$min \sum_{j=0}^{N-1} [x_{k+j}^T Q x_{k+j} + u_{k+j}^T R u_{k+j}] \quad (18)$$

The above optimization problem is solved to obtain a set of control laws out of which only the current best control action is implemented.

One would like to use an infinite horizon technique to minimize the performance objective determined by the cost However the open-loop optimal control Problem (15), that must be solved on-line, is often formulated in a finite horizon manner and the input function is parameterized finitely, in order to allow a (real-time) numerical solution of the nonlinear open-loop optimal control problem. It is clear, that the shorter the horizon, the less costly the solution of the on-line optimization problem. Thus it is desirable from a computational point of view to implement MPC schemes using short horizons. However, when a finite prediction horizon is used, the actual closed-loop input and state trajectories will differ from the predicted open-loop trajectories, even if no model plant mismatch and no disturbances are present.This fact is depicted in Figure (5) where the system can only move inside the shaded area as state constraints of the form $x(\tau) \in X$ are assumed.



Fig(5): The difference between open-loop prediction and closed-loop behavior.

This markes the key difference between standard control strategies, where the feedback law is obtained a priori and RHC where the feedback law is obtained on-line and has two immediate consequences. Firstly, the actual goal to compute a feedback such that the performance objective over the infinite horizon of the closed loop is minimized is not achieved. In general it is by no means true that a repeated minimization over a finite horizon objective in a receding horizon manner leads to an optimal solution for the infinite horizon problem.In fact, the two solutions differ significantly if a short horizon is chosen. Secondly, if the predicted and the actual trajectories differ, there is no guarantee that the closed-loop system will be stable. It is indeed easy to construct examples for which the closed-loop becomes unstable if a (small) finite horizon is chosen. Hence, when using finite horizons in standard RHC, the stage cost cannot be chosen simply based on the desired physical objectives.

C. Infnite horizon properties

Let $(x^*_\infty, u^*_\infty)(\cdot; x)$ be any optimal trajectory originating at x. Then, for any $\delta > 0$, we have

$$J^*_\infty(x^*_\infty(\delta; x)) = J^*_\infty(x) - \int_0^\delta q(x^*_\infty(\tau; x), u^*_\infty(\tau; x))d\tau.$$

$\tau \mapsto u^*_\infty(\tau; x)$ is continuous for $\tau \geq 0$, we see that

$$\lim_{\delta \to 0} \frac{J^*_\infty(x^*_\infty(\delta; x)) - J^*_\infty(x)}{\delta} = -q(x, u^*_\infty(0; x)) \leq -c_q \parallel x \parallel^2$$

so that J^*_∞ possesses a well defined (negative definite) directional derivative in each optimal direction $f(x, u^*_\infty(0; x))$.

$$J^*_\infty(x, u^*_\infty(0; x)) + q(x, u^*_\infty(0; x)) = 0$$

where $J^*_\infty(x, u)$ is the directional derivative of J^*_∞ in the direction $f(x, u)$.

Each sub-level set Γ^∞_r,r¿0,is positively invariant under optimal actions, both incremental $\delta > 0$ and infinitesimal. Also, in all cases, these sets are attracted to the origin exponentially fast. In particular, the (not necessarily unique) feedback $u = k_\infty(x) := u^*_\infty(0; x)$exponentially stabilizes the origin.

D. Finite Horizon propertities

Infinite horizon strategy can be used to provide a stabilizing feedback for a nonlinear system. One may expect a similar result while using finite horizon optimization.For instance, one could implement a receding horizon scheme as follows. From the current state x(t) obtain an optimal trajectory $(x^*_T, u^*_T)(\tau; x(t)), \tau \in [0, T]$ and use as feedback $u(t) = u^*_T(0; x(t))$.This feedback is not uniquely defined at points where more than one optimal trajectory is available. This approach requires one to continuously re-solve the finite horizon optimization. An alternative scheme is to solve the finite horizon optimization every $\delta > 0$ seconds and use the control trajectory $u^*_T(\tau; x(t)), \tau \in [0, \delta]$, to drive the system from x(t) at time t to $x^*_T(\delta, x(t))$ at time $t + \delta$.

In defining (unconstrained) finite horizon approximations to the infinite horizon problem, the key design parameters are the terminal cost function V and the horizon length T.It is well known that simple truncation of the integral (i.e. $V = 0$) may have disastrous effects if $T > 0$ is too small. Indeed, although the resulting value function may be nicely behaved, the ”optimal” receding horizon closed loop system can be unstable.

A more considered approach is to make good use of a suitable terminal cost V. Evidently, the best choice for the terminal cost is $V(x) = J^*_\infty$ since then the optimal finite and infinite horizon costs are the same. Of course, if the optimal value function were available there would be no need to solve a trajectory optimization problem.To be effective, the terminal cost must account for the discarded tail by ensuring that the origin can be reached from the terminal state $x^u(T, x)$ in an efficient manner (as measured by q). One way to do this is to use an appropriate control Lyapunov function (CLF).However, CLF is not in the scope of this paper and may be reported elsewhere. We here consider a simple unconstrained RHC strategy to control the 2D SpiderCrane.

Next, we briefly describe the steps in solving a RHC problem.

Suppose a linear, discrete-time, state-space model of the plant is given in the form

$$x(k+1) = Ax(k) + Bu(k) \quad (19)$$
$$y(k) = C_y x(k) \quad (20)$$
$$z(k) = C_z x(k) \quad (21)$$

where x is an n_x-dimensional state vector, u is an n_u-dimensional input vector,y is an n_y-dimensional vector of measured outputs and z is an n_z-dimensional vector of outputs which are to be controlled, either to particular set-points, or to satisfy some constraints, or both. The components in y and z may overlap, and may be the same that is, all the controlled outputs could as well be measured. We will assume that $y = z$, and we will then use C to denote both C_y and C_z.

We will start with the simplest situation. Assume that the whole state vector is measured, so that $\hat{x}(k|k) = x(k) =$

$y(k)(soC = I)$. Also assume that we know nothing about any disturbances or measurement noise. Then all we can do is to predict by iterating the model (19)(20). So we get

$$\begin{aligned} x(k+1|k) &= Ax(k) + Bu(k|k) \\ x(k+2|k) &= Ax(k+1|k) + Bu(k+1|k) \\ &= A^2x(k) + ABu(k|k) + Bu(k+1|k) \\ &\vdots \\ x(k+N|k) &= Ax(k+N-1|k) + Bu(k+N-1|k) \\ &= A^Nx(k) + A^{N-1}Bu(k|k) \\ &\quad + \ldots + Bu(k+N-1|k) \end{aligned}$$

which can be summarized as

$$\begin{aligned} x(k+j|k) &= A^jx(k) + \begin{bmatrix} A^{j-1} & A^{j-2} & \ldots & I \end{bmatrix} \\ &\quad \times B \begin{bmatrix} u(k|k) \\ \vdots \\ u(k+j-1|k) \end{bmatrix} \end{aligned}$$

for $j = 1, \ldots, N$.

Since we want to have the predictions expressed in terms of $\Delta u(k+j|k)$ rather than u(k + j—k) ,

$\Delta u(k+j|k) = u(k+j|k) - u(k+j-1|k)$

We use the above expression to further simplify and get the model as follows,

$$\begin{bmatrix} x(k+1|k) \\ \vdots \\ x(k+N_c|k) \\ x(k+N_c+1|k) \\ \vdots \\ x(k+N|k) \end{bmatrix} = \underbrace{\begin{bmatrix} A \\ \vdots \\ A^{N_c} \\ A^{N_c-1} \\ \vdots \\ A^N \end{bmatrix}}_{\Phi} x(k) + \underbrace{\begin{bmatrix} B & & \\ \vdots & & \\ \sum_{i=0}^{N_c-1} A^iB & & \\ \sum_{i=0}^{N_c} A^iB & & \\ \vdots & & \\ \sum_{i=0}^{N-1} A^iB & & \end{bmatrix}}_{\Gamma} u(k-1) +$$

$$\underbrace{\begin{bmatrix} B & \ldots & 0 \\ AB+B & \ldots & 0 \\ \vdots & \ddots & \vdots \\ \sum_{i=0}^{N_c-1} A^iB & \ldots & B \\ \sum_{i=0}^{N_c} A^iB & \ldots & AB+B \\ \vdots & \vdots & \vdots \\ \sum_{i=0}^{N_c-1} A^iB & \ldots & \sum_{i=0}^{N-1} A^iB \end{bmatrix}}_{G_y} \begin{bmatrix} \Delta u(k|k) \\ \vdots \\ \Delta u(k+N_c-1|k) \end{bmatrix} \tag{22}$$

The prediction of y is now obtained as

$$y(k+j|k) = Cx(k+j|k) \tag{23}$$

for $j = 1, \ldots, N$.

We can rewrite the objective function (18) as

$$J_k = \| Y(k) - Y_{ref}(k) \|_Q^2 + \| \Delta U(k) \|_R^2 \tag{24}$$

where

$$Y(k) = \begin{bmatrix} y(k+1|k) \\ \vdots \\ y(k+N|k) \end{bmatrix}, Y_{ref}(k) = \begin{bmatrix} y_{ref}(k+1|k) \\ \vdots \\ y_{ref}(k+N|k) \end{bmatrix}$$

$$\Delta U(k) = \begin{bmatrix} \Delta u(k|k) \\ \vdots \\ \Delta u(k+N_c-1|k) \end{bmatrix}$$

and the weighting matrices Q and R are given by

$$Q = \begin{bmatrix} Q(1) & 0 & \ldots & 0 \\ 0 & Q(2) & \ldots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & Q(N) \end{bmatrix}$$

$$R = \begin{bmatrix} R(0) & 0 & \ldots & 0 \\ 0 & Q(1) & \ldots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & R(N_c-1) \end{bmatrix}$$ From (22)(23) we see that Y (k) has form

$$Y(k) = \Phi x(k) + \Gamma u(k-1) + G_y \Delta U(k) \tag{25}$$

for suitable matrices $\Phi, \Gamma and G_y$. Define

$$E(k) = Y_{ref}(k) - \Phi x(k) - \Gamma u(k-1) \tag{26}$$

This vector can be thought of as a tracking error, in the sense that it is the difference between the future target trajectory and the free response of the system, namely the response that would occur over the prediction horizon if no input changes were made that is, if we set $\Delta U(k) = 0$. If E(k) really were 0,then it would indeed be correct to set $\Delta U(k) = 0$. Now we can write

$$\begin{aligned} J_k &= \| G_y\Delta U(k) - E(k) \|_Q^2 + \| \Delta U(k) \|_R^2 \\ &= [\Delta U^T(k)G_y^T - E^T(k)]Q[G_y\Delta U(k) - E(k)] \\ &\quad +\Delta U^T(k)R\Delta U(k) \\ &= \Delta U^T(k)[G_y^TQG_y + R]\Delta U(k) \\ &\quad -2E^T(k)QG_y\Delta U(k) + E^T(k)QE(k) \end{aligned} \tag{27}$$

This has the form

$$J_k = \frac{1}{2}\Delta U^T(k)H\Delta U(k) + f^T\Delta U(k) + const, \tag{28}$$

where

$H = 2(G^TyQG_y + R)$ and

$f = -2G_y^TQE(k)$

and neither H nor f depends on $\Delta U(k)$. Recall that a simple relationship exists between the input increments Δu and control input u:

$$\begin{bmatrix} u(k|k) \\ u(k+1|k) \\ \vdots \\ u(k+N_c-1|k) \end{bmatrix} = M \begin{bmatrix} \Delta u(k|k) \\ \Delta u(k+1|k) \\ \vdots \\ \Delta u(k+N_c-1|k) \end{bmatrix} + Fu(k-1),$$

where

$$M = \begin{bmatrix} I & 0 & \dots & 0 \\ I & I & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ I & I & \dots & I \end{bmatrix}, F = \begin{bmatrix} I \\ I \\ \vdots \\ I \end{bmatrix}$$

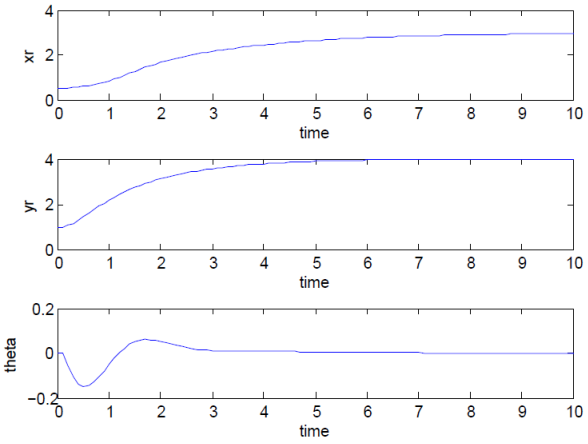
So, from (28) we see that we have to solve the following unconstrained optimization problem

$$min_{\Delta U(k)}\frac{1}{2}\Delta U^T(k)H\Delta U(k) + f^T\Delta U(k) \tag{29}$$

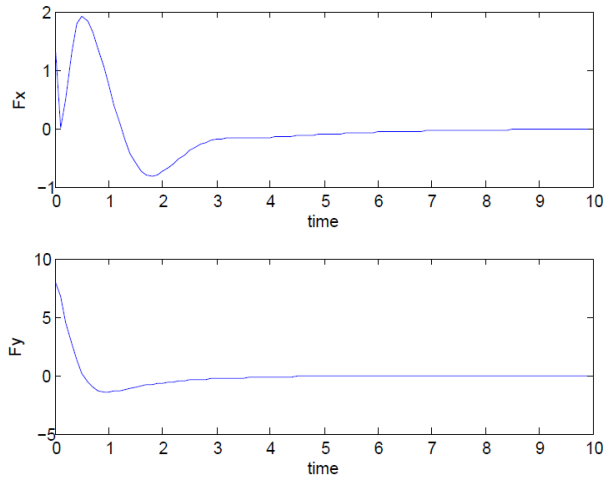
This is a standard optimization problem known as the Quadratic Programming (QP) problem, and standard algorithms are available for its solution. To solve this quadratic programming we use the quadprog tool in MATLAB to simulate our model and get the results.

IV. Simulations

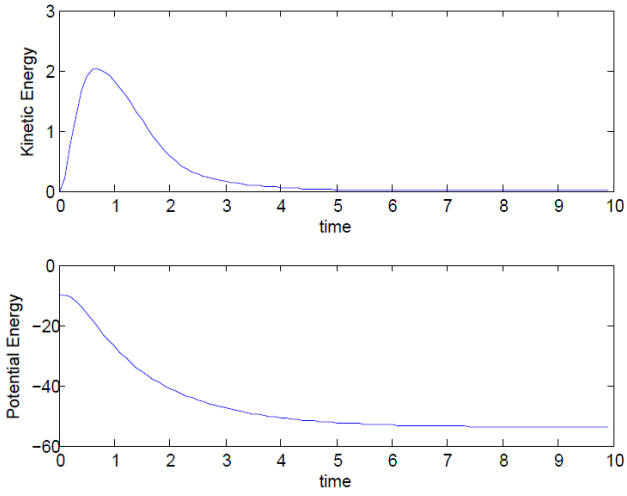
A MATLAB simulation was carried out using the following system parameters are taken as $M = 0.5$ kg for cart mass, $m = 1$ kg for payload mass and $L_3 = 0.5$ m. The performance of the controller was observed to be meeting the required objective of fast and precise payload positioning while minimizing swing.



Fig(6): Simulations results of states with Q=10,R=0.1.



Fig(7): Simulations results of control inputs with Q=10,R=0.1.



Fig(8): Simulations results of potential and kinetic energies with Q=10,R=0.1.

V. Conclusions and Future Work

As far as our knowledge goes, RHC technique has not been implemented on a 2D SpiderCrane model. Using a simple unconstrained RHC scheme, control of 2D SpiderCrane is achieved. Results from our simulation indicates that with minimum payload swing, we can achieve fast and precise payload positioning. Future work is aimed at achieving robust stabilization by combining RHC and CLF techniques.

References

[1] D. Buccieri, *Jet-scheduling Control for Flat Systems*. PhD thesis, EPFL, Lausanne, Switzerland, 2008.

[2] S. R. Oh and S. K. Agrawal, "Cable Suspended Planar Parallel Robots with Redundant Cables: Controllers with Positive Cable Tensions," *IEEE International Conference on Robotics and Automation*, vol. 3, pp. 3023–3028, September 2003.

[3] Y. Fang, W. E. Dixon, D. M. Dawson, and E. Zergeroglu, "Nonlinear Coupling Control Laws for an Underactuated Overhead Crane System ," *IEEE Trans. on Mechatronics*, vol. 8, no. 3, pp. 418–423, 2003.

[4] B. Kiss, J. Levine, and P. Mullhaupt, "A Simple Output Feedback PD Controller for Nonlinear Cranes," in *Proceedings of the 39th IEEE Conference on Decision and Control*, (sydney, Australia), pp. 5097–5101, 2000.

[5] F. Kazi, R. Banavar, R. Ortega, and N. S. Manjarekar, "Minimizing Cable Swing in a Gantry Crane Using the IDA-PBC Methodology," in *Proceedings of the 13th National Conference on Machines and Mechanisms*, (IISc, Bangalore, India), pp. 381–387, 2007.

[6] R. Banavar, F. Kazi, R. Ortega, and N. S. Manjarekar, "The IDA-PBC Methodology Applied to a Gantry Crane," in *Proceedings of the Mathematical Theory of Networks and Systems*, (Kyoto, Japan), pp. 143–147, 2006.

[7] F. Kazi, R. Banavar, R. Ortega, and N. S. Manjarekar, "Point-to-Point Control of a Gantry Crane: A Combined Flatness and IDA-PBC Strategy," in *Proceedings of the European Control Conference*, (Kos, Greece), pp. 5815–5820, 2007.

[8] M. Fliess, J. Levine, P. Martin, and P. Rouchon, "Flatness and Defect of Non-linear Systems: Introductory Theory and Examples ," *International Journal of Control*, vol. 61, no. 6, pp. 1327–1361, 1995.

[9] T. Maier and C. Woernle, "Dynamics and Control of a Cable Suspension Manipulator," in *The 9th German-Japanese seminar on Nonlinear Problems in Dynamical systems-Theory and Applications*, (Straelen, Germany), 2000.

[10] B. Kiss, J. Levine, and P. Mullhaupt, "Modelling, Flatness and Simulation of a Class of Cranes ," *Periodica Polytechnica*, vol. 43, no. 3, pp. 215–255, 1999.

[11] M. Fliess, J. Levine, P. Martin, and P. Rouchon, "A Lie-Bäcklund Approach to Equivalence and Flatness of Nonlinear Systems," *IEEE Trans. on Automatic Control*, vol. 44, no. 5, pp. 922–937, 1999.

[12] P. Martin, S. Devasia, and B. Paden, "A Different Look at Output Tracking: Control of a VTOL Aircraft," in *Proceedings of the 33th IEEE Conference on Decision and Control*, (Lake Buena Vista, FL), pp. 2376–2381, 1994.

[13] M. Rathinam and R. M. Murray, "Configuration Flatness of Lagrangian Systems Underactuated by One Control," in *Proceedings of the 35th IEEE Conference on Decision and Control*, (Kobe, Japan), pp. 1688–1693, 1996.

[14] H. Sira-Ramirez and S. K. Agrawal, *Differentially Flat Systems*. New York: Marcel Dekker, 2004.

[15] H. Sira-Ramirez, "Passivity Versus Flatness in the Regulation of an Exothermic Chemical Reactor," *European Journal of Control*, pp. 1–17, 2000.

[16] D. Buccieri, P. Mullhaupt, and D. Bonvin, "SpiderCrane Model and Properties of a Fast Weight Handling Equipment," in *The 16th IFAC world congress*, (Prague), 2005.

[17] F. Kazi, R. Banavar, P. Mullhaupt, and D. Bonvin, "Stabilization of a 2D-SpiderCrane Mechanism Using Damping Assignment Passivity-based Control," in *Proceedings of the 17th IFAC World Congress*, (Seoul, Korea), pp. 3155–3160, 2008.