# Trajectory-Based Formal Controller Synthesis for Multi-Link Robots with Elastic Joints

Sayan Saha and A. Agung Julius

***Abstract*— Multi-link robots with elastic joints are receiving a lot of interest because neglecting joint flexibility introduced in industrial robots due to presence of transmission elements results in poor control performances. Robots with elastic joints also play a pivotal role in making human-robot interaction more safe. In this paper, we discuss the problem of synthesizing provably correct controller for motion control of such robot in the presence of obstacles in the work space. For robots with many links, such task is difficult because the configuration space is high-dimensional. For example, for a robot with N links, the dimension of the configuration space is 2N, and the dimension of the state-space is 4N. To solve this problem, we built upon our previous results on trajectory-based formal controller synthesis for nonlinear systems. In this paper, we exploit the fact that the dynamics of the robot is feedback linearizable. We can then demonstrate that a provably correct controller for the robot can be obtained by using finitely many samples of valid execution trajectories. We demonstrate the validity of our results by simulating it on a multi-link robot.**

**Keywords: robots with elastic joints, trajectory robustness, feedback linearization.**

## I. Introduction

Multi-link robots with elastic joints are receiving a lot of interest in the robotics community for a number of reasons. First, arguably all joints have some flexibility. Therefore, taking this aspect into account improves the accuracy of a mathematical model of the robotic system [1], [2], [3]. Second, in advanced manufacturing, robots with flexible joints are safer as co-robots, i.e. in deployment scenarios where the robots have to have close interaction with humans [4], [5]. In this paper, we discuss the notion of formal controller synthesis for multi-link robots with elastic joints. We formulate a motion control problem for such robot in terms of a Reach/Avoid specification. This is relevant, for example, in applications where the robots need to operate in the presence of obstacles in the workspace. Earlier work in this area can be found in a comparative study between different globally stable controller strategies using backstepping and passivity based approaches, for tracking a suitable trajectory for the reduced model of a $N$-link robot arm with elastic joints [6]. A semi globally stable tracking controller and an adaptive controller, both based on implementing specific Lyapunov function candidates, for the same reduced model were presented respectively in [7] and [2]. An adaptive tracking control law based on singular perturbation was established in [8]. Designing feedforward and feedback tracking control laws using feedback linearization was explored in [9]. More recently, the focus has shifted to designing controllers for the full model with added model complexities like variable parameter values [10], [11]. However, none of this articles discuss about the generation of the suitable trajectory to accomplish the specific tasks in the Reach/Avoid situations. One might argue that path planning algorithms such as gradient based methods [12], Rapidly-exploring Random Tree (RRT) methods [13] or Probabilistic Roadmap (PRM) [14], which have been successfully employed for path planning of rigid robot arms, can be used for trajectory generation for the robots with elastic joints as well. Sub-optimal trajectory planning and tracking based on concatenating control primitives for rigid joint robots using RRT is presented in [15]. However, such methods are yet to be applied for robots with elastic joints, for which the state-space dimension is twice that of the rigid robot arms, thereby expanding the search space resulting in increasing computational complexity. Moreover, one can approximate a robot arm with flexible link as $N$ link robot arm with elastic joints, where $N$ is some large finite value [16], [17]. Indeed, increasing number of links will also add to the state-space dimension, for which the above mentioned path planning algorithms might not produce satisfactory results. It is to be noted as well that the results presented in the above mentioned articles are mostly limited to simulations of one or two link robot arms.

In this paper, we present a formal way of determining such trajectories and also guarantee a robust neighborhood around the said trajectory in the sense that the same control law can be applied for any initial state of the robot arm in the robust neighborhood. Formal controller synthesis is a very active area in the controls community. For an overview, we refer the reader to the book [18], and the references therein. The approach that we take for formal controller synthesis in this paper follows the idea presented in [19]. In short, we synthesize a provably correct controller given finitely many correct execution trajectories of the system. The idea is that these trajectories can be obtained, for example from finitely many demonstrations by a human operator. The key idea here is the assessment of safety/reachability based on the execution trajectories of the system, or the simulations thereof. To generalize the safety property of a simulated execution trajectory to a compact neighborhood around it, we use the concept of trajectory robustness [20], [21] or incremental stability [22], [23]. Roughly speaking, these properties can provide us with a bound on the divergence of the trajectories (i.e. their relative distances in $\mathcal{L}_\infty$). The main

Both the authors are with the Department of Electrical, Computer, and Systems Engineering, Rensselaer Polytechnic Institute, Troy, NY 12180, Email: sahas3,julia2@rpi.edu.

conceptual tool that is used in this approach is the theory of *approximate bisimulation*, which was developed by Girard and Pappas [24].

## II. Review of Prior Results

In this section, we briefly review prior results that were reported in [19]. Consider a multi-input multi-output dynamical system

$$\begin{aligned}\Sigma_{\text{inp}} : \frac{dx}{dt} &= f(x) + g(x)u, \\ &= \mathcal{F}(x,u),\ x \in \mathbb{R}^n,\ u \in \mathbb{R}^m, \quad (1)\\ y &= h(x),\ y \in \mathbb{R}^m. \quad (2)\end{aligned}$$

Suppose that there is a given compact set of initial states $\text{Init} \subset \mathbb{R}^n$, where the state is initiated at $t = 0$, i.e. $x(0) \in \text{Init}$. Also, we assume that there is a set of goal outputs, Goal$\subset \mathbb{R}^m$, and a set of unsafe outputs Unsafe$\subset \mathbb{R}^m$. A state trajectory $x(t)$ is deemed unsafe if its output trajectory, $y(t) = h(x(t))$, enters the unsafe set. Suppose that we are given the following control problem:

*Problem 1:* Design a feedback control law $u = k(x, x_0)$ such that for any initial state $x_0 \in \text{Init}$, the output trajectory of the closed loop system enters Goal before time $t = T_{\max}$, and remains safe until it enters Goal.

Hereafter, any (state or output) trajectory that satisfies the conditions above is called a **valid trajectory**.

*Definition 1:* A continuously differentiable function $\psi : \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}_+$ is a control autobisimulation function of (1)-(2) if for any $x, x' \in \mathbb{R}^n$,

$$\psi(x,x') \geq \|h(x) - h(x')\|, \quad (3)$$

and there exists a feedback function $k : \mathbb{R}^n \times \mathbb{R} \to \mathbb{R}^m$ such that

$$\nabla_x \psi(x,x')\mathcal{F}(x,k(x,t)) + \nabla_{x'}\psi(x,x')\mathcal{F}(x',k(x',t)) \leq 0. \quad (4)$$

If the dynamics in (1) is feedback linearizable, then we can introduce a new control input $w(t) \in \mathbb{R}^m$ and design a (nonlinear) feedback law

$$u = \kappa(x) + \lambda(x)w, \quad (5)$$

such that the closed-loop system, with the new input $w$ and output $y$, is a linear system. This is denoted by the box in Figure 1.

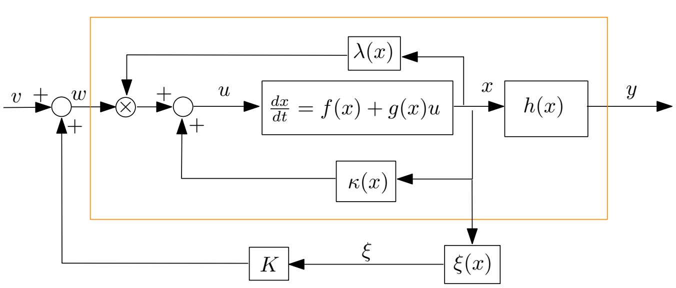


Fig. 1: Control block diagram showing the linearizing feedback and linear feedback.

Denote a minimal representation of this system as

$$\frac{d\xi}{dt} = A\xi + Bw,\ y = C\xi. \quad (6)$$

Note that there exists a (nonlinear) mapping from $x$ to $\xi$. We slightly abuse the notation and denote this mapping as $\xi(x)$. From here, we can use any linear feedback law

$$w = K\xi + v, \quad (7)$$

where $K$ is chosen such that $(A + BK)$ is Hurwitz and $v$ is a new input signal. Note that such $K$ can always be found because (6) is a minimal representation. Moreover, let $P$ be a symmetric positive definite matrix that satisfies

$$(A+BK)^T P + P(A+BK) \preceq 0, \quad (8)$$

$$P \succeq C^T C. \quad (9)$$

We can form the control bisimulation function $\psi(x, x')$ by the quadratic expression

$$\psi(x,x') \triangleq \left[(\xi(x) - \xi(x'))^T P(\xi(x) - \xi(x'))\right]^{\frac{1}{2}}. \quad (10)$$

It follows that if we define

$$k(x,t) \triangleq \kappa(x) + \lambda(x)\left(K\xi(x) + v(t)\right), \quad (11)$$

then $\psi(x, x')$ is indeed a control autobisimulation function (see Definition 1) for any choice of the control signal $v(t)$.

*Notation 1:* We denote the solution trajectory of (1) under the feedback $u = k(x, t)$, with initial state $x(0) = x_0$ as $x(t; x_0, k)$. Observing that $\psi$ defines a metric in $\mathbb{R}^n$,and we define the ball

$$B_\psi(x,r) \triangleq \{x' \mid \psi(x,x') \leq r\}. \quad (12)$$

Since the CAF is non-increasing in time (by design), we can directly deduce that for any $x_0' \in \mathbb{R}^n$ and any control signal $v(t)$,

$$x(t; x_0', k) \in B_\psi(x(t; x_0, k), \psi(x_0, x_0')).$$

This is illustrated in Figure 2. The nominal solution trajectory

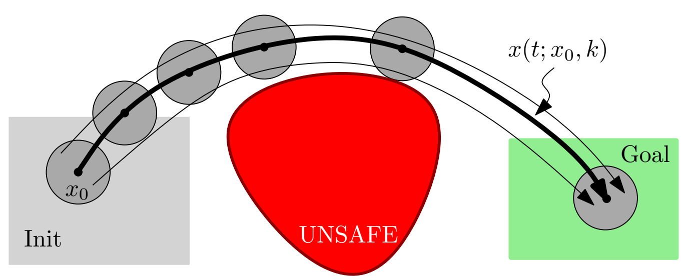


Fig. 2: Illustration of how the CAF is used in controller synthesis.

$x(t; x_0, k)$ is designed such that the condition,

$$x(T; x_0, k) \in Goal, \quad (13)$$

where, $T < T_{max}$ holds. Then we can find the robust neighborhood $B_\psi(x_0, \delta)$ around this nominal trajectory as,

$$\delta = \min(\delta_{Goal}, \delta_{Uns}), \tag{14}$$
$$\delta_{Goal} = \sup_{x' \in Goal} \psi(x(T; x_0, k), x'), \tag{15}$$
$$\delta_{Uns} = \inf_{x' \in Unsafe} \psi(x(t; x_0, k), x'). \tag{16}$$

Therefore, if an appropriate reference input signal $v(t)$ can be obtained for the nominal initial state $x_0$, it can also be used for the ball of initial conditions $B_\psi(x_0, \delta)$ around $x_0$.

**In our approach, the reference input $v(t)$ is obtained from human demonstration**. The overall idea is to obtain multiple reference input signals $v$ for multiple initial conditions, such that the corresponding "robust neighborhoods" cover the set $\mathrm{Init}$. This idea was first presented in [21] and later generalized for nonlinear systems in [19]. The objective of the current paper is to use this technique for formal controller synthesis of multi-link robots with flexible joints.

## III. Problem Formulation

### A. Dynamic Modeling of the System

Consider an open kinematic chain robot arm with $N$ rigid links, interconnected by $N$ elastic revolute joints, each of which is being actuated by an electrical motor drive. We assume that the motors are located at a position preceding the link being driven by the motor, as shown in Figure 3 for a two-link model.

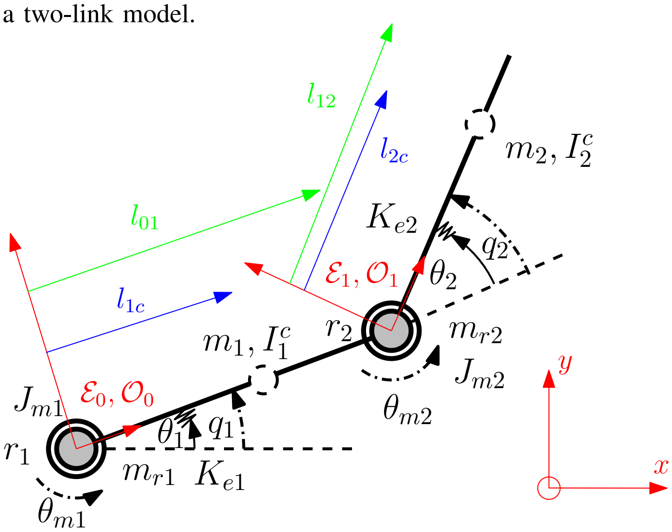


Fig. 3: Model of a two-link robot arm with elastic joints [25]

Let $q \in \mathbb{R}^N$ be the link positions, $\theta_m \in \mathbb{R}^N$ be the rotor positions of the motors, and $\theta \in \mathbb{R}^N$ be the rotor positions as reflected though the gear reduction ratios, given by $\theta_{mi} = \theta_i r_i, (i = 1, 2, \cdots, N)$, with gear ratios $r_i$.

Under the assumption that the $i$th joint deformation represented by $(q_i - \theta_i)$ is small, we can model the elastic properties of each joint to be same as that of a undamped linear torsional spring [17]. We also assume that the rotors of the motors are modeled as uniform bodies of rotation having their center of mass on the rotation axis [26]. The reduced $2N$ order dynamic model of the robot arm can then be computed to be [17]

$$\mathcal{M}(q)\ddot{q} + C(q, \dot{q})\dot{q} + g(q) + K_e(q - \theta) + F_q\dot{q} = 0, \tag{17}$$
$$\mathcal{J}\ddot{\theta} + K_e(\theta - q) + F_\theta\dot{\theta} = u. \tag{18}$$

In (17), $\mathcal{M}(q)$ represents the inertial properties of the rigid links, and $\mathcal{J}$ is a constant diagonal inertia matrix of the effective motor inertias, with the $i$th element being $\mathcal{J}_i = J_{mi} r_i^2$, where $J_{mi}$ is the inertial moment of the $i$th motor. $C(q, \dot{q})$ consists of the Christoffel symbols matrices corresponding to the Coriolis and Centrifugal terms arising from the Euler-Lagrangian dynamics. $g(q)$, is the gravity force corresponding to the potential energy of the rigid links due to gravity, and $K_e$ is a constant diagonal matrix, with the $i$th element being the $i$ joint stiffness $K_{ei}$, which we assume to be not varying. The frictional forces experienced by the links and the motors, is modeled by constant symmetric positive semidefinite matrices $F_q$ and $F_\theta$. $u \in \mathbb{R}^N$ is the column vector of the torque inputs provided by the motors defined as,

$$u = \begin{bmatrix} u_1 & u_2 & \cdots & u_N \end{bmatrix}^T. \tag{19}$$

### B. Feedback Linearization

We choose the link positions $q$ as the system output. For the state space representation we define

$$x_1 = q, \; x_2 = \theta, \; x_3 = \dot{q}, \; x_4 = \dot{\theta}$$

where, $x_i \in \mathbb{R}^N, i = (1, 2, 3, 4)$. From (17)-(18), by defining $x^T = \begin{bmatrix} x_1^T & x_2^T & x_3^T & x_4^T \end{bmatrix}$, we can write the system dynamics as

$$\dot{x} = \begin{bmatrix} x_3 \\ x_4 \\ -\mathcal{M}^{-1}(C + F_q x_3 + K_e(x_1 - x_2)) \\ -\mathcal{J}^{-1}(F_\theta x_4 + K_e(x_2 - x_1)) \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 0 \\ \mathcal{J}^{-1} \end{bmatrix} u, \tag{20}$$
$$= f(x) + g(x)u, \quad x \in \mathbb{R}^n, u \in \mathbb{R}^m, n = 4N, m = N, \tag{21}$$
$$y = x_1 = h(x), \quad y \in \mathbb{R}^m. \tag{22}$$

Here, we omitted the dependence of the variables on the system states due to space constraints in writing the equations. Notice that,

$$y^{[4]} = x_1^{[4]} = -(\mathcal{M})^{-1}((2\dot{\mathcal{M}} + F_q)x_1^{[3]} + \ddot{\mathcal{M}}\ddot{x}_1 + \ddot{\eta} + K_e(\ddot{x}_1 - \ddot{x}_2)),$$

where, $\eta(q, \dot{q}) = C(q, \dot{q})\dot{q} + g(q)$. Substituting, $\ddot{x}_2 = \dot{x}_4$ from (20) in the above equation and defining

$$\sigma(x_1, x_2) = ((2\dot{\mathcal{M}} + F_q)x_1^{[3]} + (\ddot{\mathcal{M}} + K_e)\ddot{x}_1 + \ddot{\eta}),$$

we can relate the output in terms of the input as,

$$y^{[4]} = \mathcal{M}^{-1}(K_e\mathcal{J}^{-1}(u - K_e(x_2 - x_1) - F_\theta x_4) - \sigma). \tag{23}$$

We thus conclude that the relative degree of the system is 4 and that the torque inputs given by

$$u = (\mathcal{J}K_e^{-1}M)w + (\mathcal{J}K_e^{-1}\sigma + K_e(x_2 - x_1) + F_\theta x_4), \tag{24}$$
$$= \lambda(x)w + \kappa(x), \tag{25}$$

leads to the input-output feedback linearized system in the form of a chain of integrators as

$$y^{[4]} = w, \quad w \in \mathbb{R}^N. \tag{26}$$

Also observe that, the whole state space of the system can be observed from the knowledge of the output $y$, since relative order of the system is same as the order of the actual nonlinear system, implying that the linearized system does not have any zero dynamics.

We can then obtain a minimal representation of the whole system as a set of $N$ decoupled linear subsystems with each subsystem describing the dynamics of each link-motor pair. We denote the new state-space vector $\xi$ in terms of a nonlinear mapping from $x$ to $\xi$ as $\xi^T = \begin{bmatrix} x_1^T & x_3^T & \dot{x}_3^T & \ddot{x}_3^T \end{bmatrix}^T$ The new state-space representation of the linearized subsystems then becomes

$$\frac{d\xi_i}{dt} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix} \xi_i + \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} w_i, \tag{27}$$

$$= A\xi_i + Bw_i, \quad A \in \mathbb{R}^{4\times 4}, \quad B \in \mathbb{R}^4,$$

$$y_i = q_i = C\xi_i, \quad C \in \mathbb{R}^{1\times 4}, \tag{28}$$

where, $i = (1, 2, \cdots, N)$. We design the new input $w$ in such a way that the system follows a valid trajectory, i.e, starting from any initial state in Init the end-effector of the robot arm reaches the Goal set, with no parts of the robot arm ever entering the Unsafe set.

### C. Generating Valid Trajectory

The new input $w$ is generated from the knowledge of a valid trajectory obtained using a graphical user interface provided using The Robotics Toolbox [27], that allows a human to manipulate the link positions ($q$) of a robot arm. The GUI provides the user with sliders, one for each link in the robot arm to rotate the corresponding link about its joint axis by sliding the slider left/right. This gives us a means to generate the human inputs. We fit a spline of order 5 through the initial trajectory points generated by the user and choose this new trajectory as our desired trajectory $\xi^{des}$ to be followed by the robot arm, in order to generate the **valid nominal trajectory**.

### D. Controller Design

Note from Section (II), once the system is feedback linearized, we proposed to construct a feedback law of the form

$$w_i = K_i\xi_i + v_i, \quad K_i \in \mathbb{R}^{1\times 4}. \tag{29}$$

Provided, $K_i$ is chosen such that $(A+BK_i)$ is Hurwitz, we are free to design the new input signal $v_i(t)$, such that the trajectories of the closed loop system are steered from any given initial condition in Init to be a valid trajectory.

We propose to generate $v_i(t)$ from the knowledge of the desired trajectory $\xi_i^{des}$. Note that, in the generation procedure of $\xi_i^{des}$ we do not account for the dynamics of the robot arm system, and hence $\xi_i^{des}$ cannot be considered as the valid nominal trajectory of the system itself. In order to generate the valid nominal trajectory, we design a trajectory tracking controller such that the output of the closed loop system stabilizes at the desired trajectory, which requires stabilizing $\xi_i$ at $\xi_i^{des}$. This control problem can be solved by designing a state feedback controller such that the output of each of the subsystems

$$\dot{e} = Ae + Bv_e, \quad e \in \mathbb{R}^4, \tag{30}$$

goes to zero asymptotically, where,

$$e = \xi_i - \xi_i^{des},$$

$$v_e = \xi_i^{[4]} - (\xi_i^{des})^{[4]} = w_i - (\xi_i^{des})^{[4]}. \tag{31}$$

We now propose the state feedback law of the form

$$v_e = K'e, \quad K' \in \mathbb{R}^{1\times 4}, \tag{32}$$

so that $(A+BK')$ is Hurwitz. From the requirements of (29) and (32), we can always choose $K' = K_i$. Combining (29), (31) and (32), we see that,

$$v_i = (\xi_i^{des})^{[4]} - K_i\xi_i^{des}. \tag{33}$$

It is to be taken into account at this point that, the feedback control law designed this way should belong to the class of admissible controller satisfying the control autobisimulation function (CAF) $\psi$ in order to guarantee robust neighborhoods around the valid trajectories. For each subsystem, we assume the CAF to be of the form

$$\psi(x, x') = \left[(\xi_i(x) - \xi_i(x')^T P_i(\xi_i(x) - \xi_i(x'))\right]^{\frac{1}{2}}, \tag{34}$$

where, $P_i \in \mathbb{R}^{4\times 4}$ is a symmetric positive definite matrix. We can then synthesize $P_i$ and $K_i$ by solving (8)-(9). For the whole system, the $P$ matrix is computed as $P = diag\{P_1, P_2, \cdots, P_N\}$.

## IV. Example

### A. Computing System Parameters

In order to evaluate the performance of the controller, we consider the control problem related to motion control of a 5R planar robot with elastic joints such that starting from a set of initial conditions defined by Init, the end-effector of the arm reaches a set of desired states defined by Goal, while avoiding some obstacles in the task space defined by Unsafe. We provide a general outline of the process to obtain all the necessary parameters below.

To determine the various system parameters, we follow, a coordinate frame based approach, by rigidly attaching the $i$th coordinate frame, defined as the pair $(\mathcal{E}_i, \mathcal{O}_i)$, where $\mathcal{E}_i$ consists of the unit vectors representing the $i$th coordinate frame and $\mathcal{O}_i$ is the point where the $i$th coordinate frame is being affixed, at the joint location $i$ (see Figure 3).

The inertia matrix corresponding to the inertial properties of rigid links, $\mathcal{M}$ is computed from the generalized inertia matrices $M_i, M_{ri}$ of the $i$th link-motor pair as [28]

$$\mathcal{M}(q) = \sum_{i=1}^{N} \mathbb{J}_i(q)M_i\mathbb{J}_i(q) + \sum_{i=2}^{N} \mathbb{J}_{i-1}(q)M_{ri}\mathbb{J}_{i-1}(q). \tag{35}$$

$\mathbb{J}_i$(q) is the $i$th partial Jacobian represented in the $(\mathcal{E}_i, \mathcal{O}_i)$ frame, computed as

$$\mathbb{J}_i(q) = (\vec{\mathbb{J}}_i(q))_i = \begin{bmatrix} (\vec{\mu}_1)_i & (\vec{\mu}_2)_i & \cdots & (\vec{\mu}_i)_i & 0 & \cdots & 0 \end{bmatrix},$$

where,

$$(\vec{\mu}_k)_i = \begin{bmatrix} \mathcal{R}_{i,k-1}\vec{h}_k \\ (\mathcal{R}_{i,k-1}\vec{h}_k)^{\times}(\vec{p}_{k,i})_i \end{bmatrix}.$$

The representation of the $k$th frame in $i$th frame, denoted by $\mathcal{R}_{i,k}$ is given by

$$\mathcal{R}_{i,k} = (\mathcal{R}_{i,i-1})(\mathcal{R}_{i-1,i-2})\cdots(\mathcal{R}_{k+1,k}),$$

where,

$$\mathcal{R}_{k,k+1} = I_3 + \sin(q_{k+1})(h_k)^{\times} + (1 - cos(q_{k+1}))((h_k)^{\times})^2.$$

The $(\times)$ operation on a vector results in converting it to a skew-symmetric matrix. The $k$th rotational axis is denoted by three dimensional direction vector $\vec{h}_k$, all of which for this example are coming out of the plane. $(\vec{p}_{k,i})_i$ is the geometric position vector joining $k$th frame to $i$th frame represented in $i$th frame as

$$(\vec{p}_{k,i})_i = \mathcal{R}_{i,k}p_{k,k+1} + \mathcal{R}_{i,k+1}p_{k+1,k+2} + \cdots + \mathcal{R}_{i,i-1}p_{i-1,i}.$$

where, $p_{k,k+1}$ is a constant three dimensional column vector.

The $i$th generalized link inertia matrix is calculated as

$$M_i = \begin{bmatrix} \mathcal{I}_i & m_i(p_{ic})^{\times} \\ -m_i(p_{ic})^{\times} & m_i I_3 \end{bmatrix}, \tag{36}$$

where, $\mathcal{I}_i$ is the inertia of the $i$th rigid body represented in the $i$th frame as $\mathcal{I}_i = \mathcal{I}_i^c - m_i(p_{ic})^{\times}(p_{ic})^{\times}$, where $p_{ic}$ is the geometric position vector of the center of mass of the $i$th rigid link with respect to $\mathcal{O}_i$ and $I_i^c$ is the moment of inertia of the $i$th link about its center of mass. The mass of the $i$th link is denoted by $m_i$. Similarly, the $i$th generalized motor inertia matrix is calculated as

$$M_{ri} = \begin{bmatrix} J_{mi} & m_{ri}(l_{i-1})^{\times} \\ -m_{ri}(l_{i-1})^{\times} & m_{ri} I_3 \end{bmatrix}. \tag{37}$$

$l_i$ is the $i$th link length, $m_{ri}$ represents the $i$th motor mass and $I_3$ is the third order identity matrix.

The term arising due to Coriolis and Centrifugal forces can then be computed directly from the inertia matrix $\mathcal{M}(q)$ as [28]

$$C_{i,j}(q,\dot{q}) = \tfrac{1}{2}\textstyle\sum_{k=1}^{N}\left(\frac{\partial \mathcal{M}_{i,j}}{\partial q_k} + \frac{\partial \mathcal{M}_{i,k}}{\partial q_j} - \frac{\partial \mathcal{M}_{k,j}}{\partial q_i}\right). \tag{38}$$

Since we considered a planar robot arm for this example the gravity term $g(q)$ is zero. All the above calculations were performed in MATLAB using the Symbolic Math Toolbox.

### B. Results

We simulated a 5 link robot arm to evaluate the performance of our controller design. Figure 4 illustrates the valid nominal trajectory obtained from human input for a given initial state in Init as explained in Section (III-D). The figure shows the motion of each of the 5 link tips to illustrate that no part of the robot arm ever crosses into the Unsafe and the end-effector reaches the Goal, validating that the trajectory is indeed a valid nominal trajectory of the system. Once we obtained the nominal trajectory we find the robust neighborhood around it by using (14)-(16). Figure 5 and 6 show two instances of the optimization procedure for finding the robust neighborhood around the nominal trajectory. In the figures, the robot arm represented in black corresponds

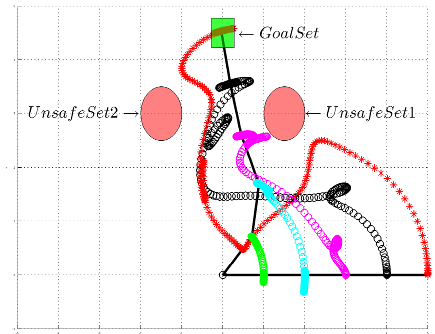


Fig. 4: Valid Nominal Trajectory obtained from human input

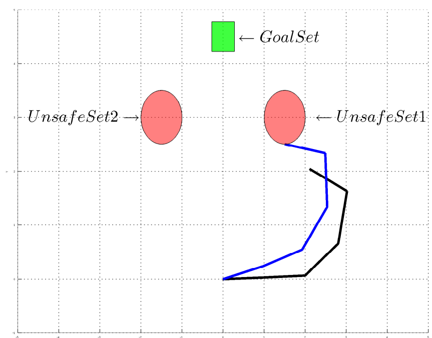


Fig. 5: Robust neighborhood around the nominal trajectory in terms of Unsafe

to the nominal trajectory, whereas, the arm in blue represents the maximum deviation allowed from the nominal trajectory, such that the resulting trajectory is still a valid one, giving us a sense of the robust neighborhood around the nominal trajectory. In the Figure 5, since the end-effector is yet to reach the Goal region, the robust neighborhood is limited by the location of the Unsafe region, however, for Figure 6 the robust neighborhood is limited by the Goal region itself. A video of the whole experiment is uploaded at http://tinyurl.com/EJrobots. An interesting thing to note in the video is that, for the nominal trajectory the robot arm displays somewhat oscillatory motion, as expected due to presence of elastic joints, unlike the rigid-body like motion of the robot arm in case of the initial trajectory generated from human inputs using the GUI, where we do not take the actual system dynamics into account.

## V. Discussion and Future Directions

We synthesize a formal trajectory based safety controller for multi-link robots with elastic joints, by utilizing our prior theoretical results which allow us to extend the concept of trajectory robustness property of linear systems to nonlinear systems if the system is feedback linearizable. The key concept presented here is to guarantee a robust neighborhood around the valid nominal trajectory obtained utilizing human intuition to solve the path planning problem regarding the motion control of the robot arm in terms of a Reach/Avoid specification. Finding such robust neighborhoods allow us to design the safety controller for the whole Init set, in terms of finitely many valid nominal trajectories.

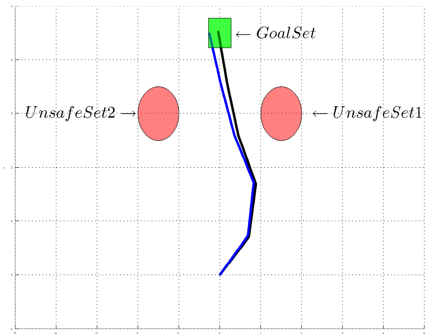

Fig. 6: Robust neighborhood around the nominal trajectory in terms of Goal

The results presented here, can be further improved by optimizing the human generated trajectory based on some desired cost function as presented in our other work [29]. Even though, we present the formulation of a formal controller utilizing the property of trajectory robustness for multi-link robots with elastic joints, the same idea can be implemented for a broader class of feedback linearizable or differentially flot nonlinear systems [19]. One particular application in this regard can be synthesizing controllers for multi-link planar manipulators actuated with 1 or 2 drives, utilizing their differential flatness property [30], [31]. A further extension of this idea can be synthesizing such safety controllers for under-actuated robot arms with elastic joints, which can then be used to approximate the behavior of robot arms with flexible links.

## VI. Acknowledgments

The research reported in this paper is partially supported by the NSF through the grants CAREER CNS-0953976 and CNS-1218109.

## References

[1] L. M. Sweet and M. Good, "Redefinition of the robot motion-control problem," *Control Systems Magazine, IEEE*, vol. 5, no. 3, pp. 18–25, 1985.

[2] R. Lozano and B. Brogliato, "Adaptive control of robot manipulators with flexible joints," *IEEE Transactions on Automatic Control*, vol. 37, no. 2, pp. 174–181, 1992.

[3] A. Albu-Schäffer and G. Hirzinger, "State feedback controller for flexible joint robots: A globally stable approach implemented on DLRs light-weight robots," in *Proc. of the 2000 IEEE/RSJ International Conf. on Intelligent Robots and Systems*, vol. 2, 2000, pp. 1087–1094.

[4] A. Albu-Schäffer, C. Ott, and G. Hirzinger, "A unified passivity-based control framework for position, torque and impedance control of flexible joint robots," *The International Journal of Robotics Research*, vol. 26, no. 1, pp. 23–39, 2007.

[5] T. Lens, J. Kunz, O. v. Stryk, C. Trommer, and A. Karguth, "Biorob-arm: A quickly deployable and intrinsically safe, light-weight robot arm for service robotics applications," in *Proc. 41st International Symposium on Robotics (ISR) and 6th German Conference on Robotics (ROBOTIK)*. VDE, 2010, pp. 1–6.

[6] B. Brogliato, R. Ortega, and R. Lozano, "Global tracking controllers for flexible-joint manipulators: a comparative study," *Automatica*, vol. 31, no. 7, pp. 941–956, 1995.

[7] A. Loria and R. Ortega, "On tracking control of rigid and flexible joints robots," *Appl. Math. Comput. Sci*, vol. 5, no. 2, pp. 101–113, 1995.

[8] C. Ott, A. Albu-Schaffer, and G. Hirzinger, "Comparison of adaptive and nonadaptive tracking control laws for a flexible joint manipulator," in *Proc. IEEE/RSJ International Conference on Intelligent Robots and Systems*, vol. 2. IEEE, 2002, pp. 2018–2024.

[9] A. De Luca, "Feedforward/feedback laws for the control of flexible robots," in *Proc. IEEE International Conference on Robotics and Automation*, vol. 1. IEEE, 2000, pp. 233–240.

[10] G. Palli, C. Melchiorri, and A. De Luca, "On the feedback linearization of robots with variable joint stiffness," in *Proc. IEEE International Conference on Robotics and Automation*. IEEE, 2008, pp. 1753–1759.

[11] L. Zollo, B. Siciliano, A. De Luca, E. Guglielmelli, and P. Dario, "Compliance control for an anthropomorphic robot with elastic joints: Theory and experiments," *Journal of dynamic systems, measurement, and control*, vol. 127, no. 3, pp. 321–328, 2005.

[12] N. Ratliff, M. Zucker, J. A. Bagnell, and S. Srinivasa, "CHOMP: Gradient optimization techniques for efficient motion planning," in *Proc. IEEE International Conference on Robotics and Automation*. IEEE, 2009, pp. 489–494.

[13] J. J. Kuffner and S. M. LaValle, "RRT-connect: An efficient approach to single-query path planning," in *Proc. IEEE International Conference on Robotics and Automation*, vol. 2. IEEE, 2000, pp. 995–1001.

[14] L. E. Kavraki, P. Svestka, J.-C. Latombe, and M. H. Overmars, "Probabilistic roadmaps for path planning in high-dimensional configuration spaces," *Robotics and Automation, IEEE Transactions on*, vol. 12, no. 4, pp. 566–580, 1996.

[15] M. Steinegger, B. Passenberg, M. Leibold, and M. Buss, "Trajectory planning for manipulators based on the optimal concatenation of LQ control primitives," *Decision and Control and European Control Conference (CDC-ECC), 50th IEEE Conference on*, pp. 2837–2842, 2011.

[16] P. Tomei and A. Tornambe, "Approximate modeling of robots having elastic links," *Systems, Man and Cybernetics, IEEE Transactions on*, vol. 18, no. 5, pp. 831–840, 1988.

[17] P. Tomei, "A simple PD controller for robots with elastic joints," *Automatic Control, IEEE Transactions on*, vol. 36, no. 10, pp. 1208–1213, 1991.

[18] P. Tabuada, *Verification and control of hybrid systems: a symbolic approach*. Springer, 2009.

[19] A. A. Julius and A. K. Winn, "Safety controller synthesis using human generated trajectories: Nonlinear dynamics with feedback linearization and differential flatness," in *Proc. American Control Conference*, Montreal, Canada., 2012, pp. 709–714.

[20] A. A. Julius, "Trajectory-based controller design for hybrid systems with affine continuous dynamics," in *Proc. IEEE Conf. Automation Science and Engineering*, Toronto, Canada, 2010, pp. 1007–1012.

[21] A. A. Julius and S. Afshari, "Using computer games for hybrid systems controller synthesis," in *Proc. 49th IEEE Conf. Decision and Control*, Atlanta, Georgia, 2010, pp. 5887–5892.

[22] P. Tabuada, "An approximate simulation approach to symbolic control," *IEEE Trans. Automatic Control*, vol. 53, no. 6, pp. 1406–1418, 2008.

[23] G. Pola, A. Girard, and P. Tabuada, "Approximately bisimilar symbolic models for nonlinear control systems," *Automatica*, vol. 44, no. 10, pp. 2508–2516, 2008.

[24] A. Girard and G. J. Pappas, "Approximation metrics for discrete and continuous systems," *IEEE Trans. Automatic Control*, vol. 52, no. 5, pp. 782–798, 2007.

[25] A. De Luca and P. Lucibello, "A general algorithm for dynamic feedback linearization of robots with elastic joints," in *Proc. IEEE International Conference on Robotics and Automation*, vol. 1. IEEE, 1998, pp. 504–510.

[26] M. W. Spong, "Modeling and control of elastic joint robots," *Journal of dynamic systems, measurement, and control*, vol. 109, no. 4, pp. 310–318, 1987.

[27] P. I. Corke, *Robotics, Vision & Control: Fundamental Algorithms in Matlab*. Springer, 2011.

[28] R. M. Murray, Z. Li, and S. S. Sastry, *A mathematical introduction to robotic manipulation*. CRC press, 1994.

[29] A. K. Winn and A. A. Julius, "Optimization of human generated trajectories for safety controller synthesis," in *Proc. American Control Conference (ACC), 2013*. IEEE, 2013, pp. 4374–4379.

[30] S. K. Agrawal and V. Sangwan, "Differentially flat designs of under-actuated open-chain planar robots," *Robotics, IEEE Transactions on*, vol. 24, no. 6, pp. 1445–1451, 2008.

[31] J. Franch, S. K. Agrawal, and V. Sangwan, "Differential flatness of a class of n-DOF planar manipulators driven by 1 or 2 actuators," *Automatic Control, IEEE Transactions on*, vol. 55, no. 2, pp. 548–554, 2010.