A Barrier Pair Method for Safe Human-Robot Shared Autonomy

Binghan He[1], Mahsa Ghasemi, Ufuk Topcu and Luis Sentis

***Abstract*— Shared autonomy provides a framework where a human and an automated system, such as a robot, jointly control the system's behavior, enabling an effective solution for various applications, including human-robot interaction. However, a challenging problem in shared autonomy is safety because the human input may be unknown and unpredictable, which affects the robot's safety constraints. If the human input is a force applied through physical contact with the robot, it also alters the robot's behavior to maintain safety. We address the safety issue of shared autonomy in real-time applications by proposing a two-layer control framework. In the first layer, we use the history of human input measurements to infer what the human wants the robot to do and define the robot's safety constraints according to that inference. In the second layer, we formulate a rapidly-exploring random tree of barrier pairs, with each barrier pair composed of a barrier function and a controller. Using the controllers in these barrier pairs, the robot is able to maintain its safe operation under the intervention from the human input. This proposed control framework allows the robot to assist the human while preventing them from encountering safety issues. We demonstrate the proposed control framework on a simulation of a two-linkage manipulator robot.**

I. Introduction

Unlike full robot autonomy, shared autonomy allows a robot to leverage the perceptual and decision making capabilities of operators while helping them to work more efficiently and accurately [1]. Across different fields, such as brain-computer interfaces [2], autonomous driving [3], and teleoperation [4], shared autonomy helps us to improve our productivity without completely removing the human from the task at hand. However, safety becomes critical with shared autonomy, especially when operators and robots interact through physical contact. On the one hand, the human's objective is not directly measurable but can be inferred based on the robot's sensing of human inputs such as contact forces. The robot needs this inference of the human's objective to figure out how to assist the human and prevent them from potential accidents. On the other hand, human inputs can alter the robot's current path resulting in additional safety concerns. Therefore in a shared autonomy task, the robot faces a conflict between inferring the human's objectives and maintaining safety under the interaction with human inputs.

For a nonlinear dynamical system such as a robot, safety is usually verified through barrier functions [5]. Just like Lyapunov functions for stability verification, barrier functions provide sufficient conditions for safety verification. But barrier functions relax the global convergence requirement of Lyapunov functions and only need to be decreasing at the safety bounds. Various methods create barrier functions with controllers to enforce safety constraint satisfaction. For state-space constraints, controllers can be synthesized simultaneously with barrier functions using back-stepping [6] or quadratic programming methods [7], [8]. Input constraints can also be enforced using semi-definite programming methods [9], [10]. By incorporating sampling-based methods into the synthesis of barrier functions and controllers [11], robots can also guarantee safe operation with non-convex state-space constraints.

While the above methods aim to resolve the safety problem for a robot alone, it is a more challenging problem to guarantee safety for a robot that has physical contact with a person. This is because humans represent an uncertain dynamical sub-system when physically interacting with robots. The internal states of the human dynamics are usually immeasurable. Robust control strategies [12], [13] can address the uncertain human dynamics and achieve complementary stability for human-robot coupled system. However, this kind of strategy usually considers the robot as a strict follower of the human's trajectory and hence, relies on the human to obey safety constraints. In human-robot shared autonomy, the robot needs to enforce safety constraints in relation to the human's objective such that it can prevent them from potential accidents.

In this paper, we address the safety problem during shared autonomy using a two-layer control framework. In the first layer, we define the robot's safety constraints based on the inference of the human's objective. In order to understand the human's objective, we propose an intent inference method that integrates the history of human's input, obtained through the sensor readings, to generate a probability distribution over a set of candidate objectives. In particular, the proposed method employs a Boltzmann model of rationality [14], [15] to characterize the probability of receiving a particular sensor reading at the robot's end-effector based on the human's objective. Using this Boltzmann model, the robot keeps track of a probability distribution, called belief, over the candidate objectives and updates it using a Bayesian method. Then, the belief is handed to a safety controller in the second layer so that the robot can safely move toward the most probable human objective.

In the second layer, we use the barrier pair rapidly-exploring random tree method [11] to generate sequences of barrier pairs given different human objectives. Each barrier

The authors are with the Department of Mechanical Engineering (B.H.), the Department of Electrical and Computer Engineering (M.G.) and the Department of Aerospace Engineering and Engineering Mechanics (U.T., L.S.), The University of Texas at Austin, Austin, TX. Send correspondence to [1] binghan at utexas dot edu.

pair comprises a quadratic barrier function and a state feedback controller. We synthesize the state feedback controllers in these barrier pairs using a robust control strategy so that the robot can satisfy the safety constraints for different human objectives and reject the human input interventions. Based on the human intention inference formulated from the first layer, the robot can execute these barrier pair sequences accordingly and help the human to safely accomplish the objective. We demonstrate this two-layer control framework on a simulation of a two-linkage manipulator robot, where a human operator uses a keyboard to control a simulated human force exerted on the end-effector of the manipulator robot.

II. Preliminaries

In this section, we first overview the basics of multi-body robot dynamics and barrier pair rapidly-exploring random trees. Then, we present the formal problem statement. a_i is defined as a polytopic region in the workspace of a robot. For convenience, $\mathbf{x}_{\mathsf{a}_\mathsf{i}}$ is defined as the geometric center for the region of a_i and $\bar{\mathsf{a}}_\mathsf{i} \triangleq \mathbb{R}^\mathrm{n} \setminus \mathsf{a}_\mathsf{i}$ is defined as a workspace region excluding the set for a_i.

A. Multi-Body Robot Dynamics

The Lagrangian dynamics of an n-DOF robot can be expressed as

$$\mathbf{M}(\mathbf{q}) \cdot \ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}}) \cdot \dot{\mathbf{q}} = \mathbf{u} + \mathbf{J}^\top(\mathbf{q}) \cdot \mathbf{w} \tag{1}$$

where $\mathbf{M}(\mathbf{q})$ is the matrix of inertia, $\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})$ is the coefficient matrix of Coriolis and centrifugal effects, $\mathbf{J}(\mathbf{q})$ is the matrix of Jacobian, $\mathbf{q} \triangleq [q_1, \cdots, q_\mathrm{n}]^\top$ is the vector of joint positions with $\dot{\mathbf{q}}$ and $\ddot{\mathbf{q}}$ defined as its first and second order time derivatives, $\mathbf{u} \triangleq [u_1, \cdots, u_\mathrm{n}]^\top$ is the vector of joint torques and $\mathbf{w} \triangleq [w_1, \cdots, w_\mathrm{n}]^\top$ is the vector of external forces exerted by the human. An n-dimensional workspace position vector $\mathbf{x} \triangleq [x_1, \cdots, x_\mathrm{n}]^\top$ can be calculated from the joint position vector using

$$\mathbf{x} = \mathbf{F}(\mathbf{q}) \tag{2}$$

where $\mathbf{F}(\cdot)$ represents the forward kinematics. By linearizing (1) and (2) around an equilibrium point $[\mathbf{q_e}^\top, \vec{0}^\top]^\top$, we obtain the state-space form

$$\begin{bmatrix}\dot{\tilde{\mathbf{q}}} \\ \ddot{\tilde{\mathbf{q}}}\end{bmatrix} = \begin{bmatrix}\mathbf{0} & \mathbf{I} \\ \mathbf{0} & \mathbf{M}^{-1}(\mathbf{q_e}) \cdot \mathbf{C}(\mathbf{q_e}, \vec{0})\end{bmatrix} \begin{bmatrix}\tilde{\mathbf{q}} \\ \dot{\tilde{\mathbf{q}}}\end{bmatrix} + \begin{bmatrix}\mathbf{0} \\ \mathbf{M}^{-1}(\mathbf{q_e})\end{bmatrix} \mathbf{u} + \begin{bmatrix}\mathbf{0} \\ \mathbf{M}^{-1}(\mathbf{q_e}) \cdot \mathbf{J}^\top(\mathbf{q_e})\end{bmatrix} \mathbf{w} \tag{3}$$

$$\tilde{\mathbf{x}} = \begin{bmatrix}\mathbf{J}(\mathbf{q_e}) & \mathbf{0}\end{bmatrix} \begin{bmatrix}\tilde{\mathbf{q}} \\ \dot{\tilde{\mathbf{q}}}\end{bmatrix} \tag{4}$$

where $\tilde{\mathbf{q}} \triangleq \mathbf{q} - \mathbf{q_e}$ and $\tilde{\mathbf{x}} \triangleq \mathbf{x} - \mathbf{x_e}$ with $\mathbf{x_e} = \mathbf{F}(\mathbf{q_e})$. The partial derivative of $\mathbf{F}(\mathbf{q})$ with respect to $\mathbf{q}$ is the Jacobian matrix $\mathbf{J}(\mathbf{q})$.

Algorithm 1 $\mathbf{G} \leftarrow \mathtt{BPRRT}(\mathsf{a}_0, \mathsf{a}_\mathsf{f}, \bar{\mathsf{a}}_1, \cdots, \bar{\mathsf{a}}_{\mathrm{n}_\mathrm{o}}, \mathsf{Z}_0, \mathsf{U}, \varepsilon)$

Input: Initial region a_0, goal region a_f, constraints associated with undesirable regions $\bar{\mathsf{a}}_1, \cdots, \bar{\mathsf{a}}_{\mathrm{n}_\mathrm{o}}$, state space constraint Z_0, input constraint U, scalar ε $(0 < \varepsilon \leq 1)$

Output: BP-RRT graph $\mathbf{G}$

$(\mathbf{Q}_\mathsf{f}, \mathbf{K}_\mathsf{f}) \leftarrow \mathtt{BP}(\mathbf{x}_{\mathsf{a}_\mathsf{f}}, \mathsf{a}_\mathsf{f}, \bar{\mathsf{a}}_1, \cdots, \bar{\mathsf{a}}_{\mathrm{n}_\mathrm{o}}, \mathsf{Z}_0, \mathsf{U})$
$\mathbf{G}.\mathtt{AddVertex}(\mathbf{x}_\mathsf{f})$, $\mathbf{G}.\mathtt{AddBP}((\mathbf{Q}_\mathsf{f}, \mathbf{K}_\mathsf{f}))$
$(\mathbf{Q}_\mathsf{new}, \mathbf{K}_\mathsf{new}) \leftarrow (\mathbf{Q}_\mathsf{f}, \mathbf{K}_\mathsf{f})$, $\mathbf{x}_\mathsf{new} \leftarrow \mathbf{x}_\mathsf{f}$
while $\mathbf{x}_0 \notin \mathbf{E}_\mathsf{new}(\varepsilon)$ **do**
  $\mathbf{q}_\mathsf{rand} \leftarrow \mathtt{RandomConfiguration}(\bigcap_{\mathsf{i}=1}^{\mathrm{n}_\mathrm{o}} \bar{\mathsf{a}}_\mathsf{i})$
  $\mathbf{E}_\mathsf{near}(\varepsilon) \leftarrow \mathtt{NearestBP}(\mathbf{q}_\mathsf{rand}, \mathbf{G}, \varepsilon)$
  $\mathbf{q}_\mathsf{new} \leftarrow \mathtt{NewEquilibrium}(\mathbf{q}_\mathsf{rand}, \mathbf{E}_\mathsf{near}(\varepsilon))$
  $\mathbf{x}_\mathsf{new} \leftarrow \mathbf{F}(\mathbf{q}_\mathsf{new})$
  $(\mathbf{Q}_\mathsf{new}, \mathbf{K}_\mathsf{new}) \leftarrow \mathtt{BP}(\mathbf{x}_\mathsf{new}, \emptyset, \bar{\mathsf{a}}_1, \cdots, \bar{\mathsf{a}}_{\mathrm{n}_\mathrm{o}}, \mathsf{Z}_0, \mathsf{U})$
  $\mathbf{G}.\mathtt{AddVertex}(\mathbf{x}_\mathsf{new})$, $\mathbf{G}.\mathtt{AddBP}((\mathbf{Q}_\mathsf{new}, \mathbf{K}_\mathsf{new}))$, $\mathbf{G}.\mathtt{AddEdge}((\mathbf{x}_\mathsf{near}, \mathbf{x}_\mathsf{new}))$
end while
$(\mathbf{Q}_0, \mathbf{K}_0) \leftarrow \mathtt{BP}(\mathbf{x}_{\mathsf{a}_0}, \mathsf{a}_0, \bar{\mathsf{a}}_1, \cdots, \bar{\mathsf{a}}_{\mathrm{n}_\mathrm{o}}, \mathsf{Z}_0, \mathsf{U})$
$\mathbf{G}.\mathtt{AddVertex}(\mathbf{x}_0)$, $\mathbf{G}.\mathtt{AddBP}((\mathbf{Q}_0, \mathbf{K}_0))$, $\mathbf{G}.\mathtt{AddEdge}((\mathbf{x}_\mathsf{new}, \mathbf{x}_0))$

B. Barrier Pair Rapidly-Exploring Random Trees

Definition 1 [10]: A *barrier pair* is a pair consisting of a barrier function and a controller $(\mathbf{B},\ \mathbf{k})$ with the following properties

(a) $-1 < \mathbf{B}(\tilde{\mathbf{q}}, \dot{\tilde{\mathbf{q}}}) \leq 0, \mathbf{u} = \mathbf{k}(\tilde{\mathbf{q}}, \dot{\tilde{\mathbf{q}}}) \implies \dot{\mathbf{B}}(\tilde{\mathbf{q}}, \dot{\tilde{\mathbf{q}}}) < 0,$
(b) $\mathbf{B}(\tilde{\mathbf{q}}, \dot{\tilde{\mathbf{q}}}) \leq 0 \implies [\tilde{\mathbf{q}}^\top, \dot{\tilde{\mathbf{q}}}^\top]^\top \in \mathsf{Z},\ \mathbf{k}(\tilde{\mathbf{q}}, \dot{\tilde{\mathbf{q}}}) \in \mathsf{U},$

where $[\tilde{\mathbf{q}}^\top, \dot{\tilde{\mathbf{q}}}^\top]^\top \in \mathsf{Z}$ and $\mathbf{u} \in \mathsf{U}$ are the state and input constraints.

If we define the barrier pair as

$$\mathbf{B} = \begin{bmatrix}\tilde{\mathbf{q}} \\ \dot{\tilde{\mathbf{q}}}\end{bmatrix}^\top \mathbf{Q}^{-1} \begin{bmatrix}\tilde{\mathbf{q}} \\ \dot{\tilde{\mathbf{q}}}\end{bmatrix} - 1, \quad \mathbf{k} = \mathbf{K} \begin{bmatrix}\tilde{\mathbf{q}} \\ \dot{\tilde{\mathbf{q}}}\end{bmatrix} \tag{5}$$

where $\mathbf{B}$ is a quadratic barrier function with a positive definite matrix $\mathbf{Q}$ and $\mathbf{k}$ is a full state feedback controller, the barrier pair synthesis becomes a linear matrix inequality (LMI) optimization problem [10].

For convenience, we use $(\mathbf{Q}, \mathbf{K})$ to represent a barrier pair $(\mathbf{B}, \mathbf{k})$ in the form of (5) and define $\mathbf{E}(\varepsilon) \triangleq \{[\tilde{\mathbf{q}}^\top, \dot{\tilde{\mathbf{q}}}^\top]^\top \mid [\tilde{\mathbf{q}}^\top, \dot{\tilde{\mathbf{q}}}^\top]\mathbf{Q}^{-1}[\tilde{\mathbf{q}}^\top, \dot{\tilde{\mathbf{q}}}^\top]^\top \leq \varepsilon^2\}$ as the sub-level set of $\mathbf{B}$ corresponding to a value $\varepsilon^2 - 1$. Based on Definition 1, the zero sub-level set $\mathbf{E}(1)$ of the barrier function $\mathbf{B}$ needs to satisfy all constraints defined by Z and U.

In [11], a barrier pair rapidly-exploring random tree (BP-RRT) method is introduced which leverages rapidly-

exploring random trees (RRT) to combine a number of barrier pairs into a sequence that connects two polytopic regions in the reachable workspace. Compared RRT, the BP-RRT adds a barrier pair to each vertex in a graph providing additional robustness and safety guarantees for trajectory execution.

Algorithm 1 shows the procedure for creating a BP-RRT graph. Instead of applying a fixed incremental distance as RRT does in each iteration, a new robot configuration $\mathbf{q}_{\mathsf{new}}$ is added to the graph by projecting a random configuration $\mathbf{q}_{\mathsf{rand}}$ to the hyper-surface of the sub-level set $\mathbf{E}_{\mathsf{near}}(\varepsilon)$ of the nearest barrier pair (for $0 < \varepsilon \leq 1$). Therefore, $\mathbf{q}_{\mathsf{new}}$ is guaranteed to be inside the zero sub-level set of previously created barrier pairs. The algorithm terminates if there exists a new barrier pair $(\mathbf{Q}_0, \mathbf{K}_0)$ in the BP-RRT graph whose zero sub-level set $\mathbf{E}_0(1)$ contains the entire region of a_0. Then, a sequence of barrier pairs that connects a_0 and a_f can be extracted from the BP-RRT graph.

C. Problem Statement

In this paper, we consider a robot operating around multiple different polytopic regions defined in the workspace of its end-effector and a human operator that applies a norm-bound interaction force to the robot's end-effector intermittently.

Problem: During real-time human-robot shared autonomy operation, we aim to infer the operator's target region from a time series of intermittent human force measurements and create a sequence of barrier pairs such that the robot's end-effector can safely move to the target region without passing through all other regions.

III. Barrier Pair Synthesis

Similar to [11], our process of barrier pair synthesis starts by linearizing the robot's dynamics in (3) and (4) such that a norm-bound linear differential inclusion (LDI) model can be formulated. Then, a LMI optimization problem can be created for synthesizing barrier pairs subject to predefined state space and input constraints. In particular, we formulate a LMI constraint in the barrier pair synthesis problem for enforcing the robot's stability and the convergence of the barrier function under the influence of the norm-bound force input $\mathbf{w}$ from the human.

A. Norm-Bound Linear Differential Inclusion Model

Our barrier pair synthesis relies on solving an LMI optimization problem formulated based on a linear model of the robot dynamics. However, the linearized state space equations in (3) and (4) become inaccurate if the state $[\mathrm{q}^\top,\, \dot{\mathrm{q}}^\top]^\top$ deviates from the equilibrium. In order to address this issue, we use a norm-bound LDI to represent the robot dynamics. First, we can express the norm-bound uncertainties of the linearized robot dynamical model in (3) and (4) as

$$\mathbf{M}^{-1}(\mathbf{q}) \cdot \mathbf{C}(\mathbf{q},\, \dot{\mathbf{q}}) \in \{\mathbf{A}_1 + \mathbf{A}_2 \Delta \mathbf{A}_3 :\ \|\Delta\| \leq 1\} \tag{6}$$

$$\mathbf{M}^{-1}(\mathbf{q}) \cdot \mathbf{J}^\top(\mathbf{q}) \in \{\mathbf{B}_1^{\mathbf{w}} + \mathbf{B}_2^{\mathbf{w}} \Delta \mathbf{B}_3^{\mathbf{w}} :\ \|\Delta\| \leq 1\} \tag{7}$$

$$\mathbf{M}^{-1}(\mathbf{q}) \in \{\mathbf{B}_1^{\mathbf{u}} + \mathbf{B}_2^{\mathbf{u}} \Delta \mathbf{B}_3^{\mathbf{u}} : \|\Delta\| \leq 1\} \tag{8}$$

$$\mathbf{J}(\mathbf{q}) \in \{\ \mathbf{J}_1 + \mathbf{J}_2 \Delta \mathbf{J}_3\ :\ \|\Delta\| \leq 1\} \tag{9}$$

for all state $[\mathbf{q}^\top,\, \dot{\mathbf{q}}^\top]^\top$ in the constrained state space Z around the equilibrium. Then, a norm-bound LDI [16] that is valid for all states in Z can be expressed as

$$\begin{bmatrix} \dot{\tilde{\mathbf{q}}} \\ \ddot{\tilde{\mathbf{q}}} \end{bmatrix} = \begin{bmatrix} \mathbf{0} & \mathbf{I} \\ \mathbf{0} & \mathbf{A}_1 + \mathbf{A}_2 \Delta \mathbf{A}_3 \end{bmatrix} \begin{bmatrix} \tilde{\mathbf{q}} \\ \dot{\tilde{\mathbf{q}}} \end{bmatrix} + \begin{bmatrix} \mathbf{0} \\ \mathbf{B}_1^{\mathbf{u}} + \mathbf{B}_2^{\mathbf{u}} \Delta \mathbf{B}_3^{\mathbf{u}} \end{bmatrix} \mathbf{u} + \begin{bmatrix} \mathbf{0} \\ \mathbf{B}_1^{\mathbf{w}} + \mathbf{B}_2^{\mathbf{w}} \Delta \mathbf{B}_3^{\mathbf{w}} \end{bmatrix} \mathbf{w} \tag{10}$$

$$\tilde{\mathbf{x}} = \begin{bmatrix} \mathbf{J}_1 + \mathbf{J}_2 \Delta \mathbf{J}_3 & \mathbf{0} \end{bmatrix} \begin{bmatrix} \tilde{\mathbf{q}} \\ \dot{\tilde{\mathbf{q}}} \end{bmatrix}. \tag{11}$$

We can formulate a norm-bound LDI by calculating $\mathbf{M}^{-1}(\mathbf{q}) \cdot \mathbf{C}(\mathbf{q},\, \dot{\mathbf{q}})$, $\mathbf{M}^{-1}(\mathbf{q}) \cdot \mathbf{J}^\top(\mathbf{q},\, \dot{\mathbf{q}})$, $\mathbf{M}^{-1}(\mathbf{q})$ and $\mathbf{J}(\mathbf{q})$ from a number of sample states in Z and using quadric inclusion programs [17] to fit an inclusion model.

The constrained state space region for the norm-bound LDI is defined as $\mathsf{Z} \triangleq \mathsf{Z}_{\mathsf{safe}} \cap \mathsf{Z}_0$. Based on the inequality constraints $|\mathbf{a}_\mathrm{i} \tilde{\mathbf{x}}| < \bar{a}_\mathrm{i}$ associated with the undesirable regions $\mathsf{a}_1, \mathsf{a}_2, \cdots, \mathsf{a}_{\mathrm{n_o}}$, a local convex state space region $\mathsf{Z}_{\mathsf{safe}}$ can be defined as

$$\begin{aligned} \mathsf{Z}_{\mathsf{safe}} \triangleq \{[\tilde{\mathbf{q}}^\top,\, \dot{\tilde{\mathbf{q}}}^\top]^\top : |\mathbf{a}_\mathrm{i}(\mathbf{J}_1 + \mathbf{J}_2 \Delta \mathbf{J}_3)\ \tilde{\mathbf{q}}| < \bar{a}_\mathrm{i}, \\ \|\Delta\| \leq 1,\ \mathrm{i} = 1, \cdots, \mathrm{n_o}\}, \end{aligned} \tag{12}$$

where $\mathbf{a}_\mathrm{i}$ for $\mathrm{i} = 1, \cdots, \mathrm{n_o}$ are row vectors with $\mathrm{n_o}$ as the number of undesirable regions. However, the norm-bound uncertainty in $\mathsf{Z}_{\mathsf{safe}}$ can be too large for the barrier pair sub-problem to be solved. So we also need to consider an additional constrained state space Z_0 defined as

$$\begin{aligned} \mathsf{Z}_0 \triangleq \{[\tilde{\mathbf{q}}^\top,\, \dot{\tilde{\mathbf{q}}}^\top]^\top : |\mathbf{b}_\mathrm{i}(\mathbf{J}_1 + \mathbf{J}_2 \Delta \mathbf{J}_3)\ \tilde{\mathbf{q}}| < \bar{x}_\mathrm{i},\ |\mathbf{b}_\mathrm{i} \dot{\tilde{\mathbf{q}}}| < \bar{\dot{q}}_\mathrm{i}, \\ \|\Delta\| \leq 1,\ \mathrm{i} = 1, \cdots, \mathrm{n}\}, \end{aligned} \tag{13}$$

where $\mathbf{b}_\mathrm{i}$ for $\mathrm{i} = 1, \cdots, \mathrm{n}$ are the standard basis (row) vectors of the n-dimensional Euclidean space.

Similar to (12) and (13), a constrained input space region U and a constrained external input space region W can be formulated as

$$\mathsf{U} \triangleq \{\mathbf{u} : |\mathbf{b}_\mathrm{i} \mathbf{u}| < \bar{u}_\mathrm{i},\ \mathrm{i} = 1, \cdots, \mathrm{n}\}. \tag{14}$$

$$\mathsf{W} \triangleq \{\mathbf{w} : \|\mathbf{w}\| < \bar{w}\}. \tag{15}$$

B. Barrier Pair Synthesis Sub-Problems

Based on the norm bound LDI model expressed in (10) and (11), we can formulate the LMIs for creating our barrier pair synthesis problem. In order to ensure that the ellipsoidal sub-level set $\mathbf{E}(1)$ of a barrier pair contains a desired polytopic region a_d, we sample a number of points from all edges of a_d and let $\mathbf{E}(1)$ contain the joint space projections of these Cartesian space samples using the following set of LMIs

$$\begin{bmatrix} 1 & \star \\ \mathbf{R}(\mathbf{x}_\mathrm{i}) - \mathbf{q}_\mathbf{e} & \mathbf{S}_1 \mathbf{Q} \mathbf{S}_1^\top \end{bmatrix} \succeq 0, \quad \forall\, \mathrm{i} = 1, \ldots, \mathrm{n_p} \tag{16}$$

where $\mathrm{n_p}$ is the number of sampled workspace points at the edge of a_d, $\mathsf{a}_\mathsf{d} = \mathsf{Co}\{\mathbf{x}_1, \cdots, \mathbf{x}_\mathrm{p}\}$, $\mathbf{R}(\cdot)$ is an inverse kinematics operator and $\mathbf{S}_1 \triangleq [\mathbf{I}_{\mathrm{n} \times \mathrm{n}},\, \mathbf{0}_{\mathrm{n} \times \mathrm{n}}]$.

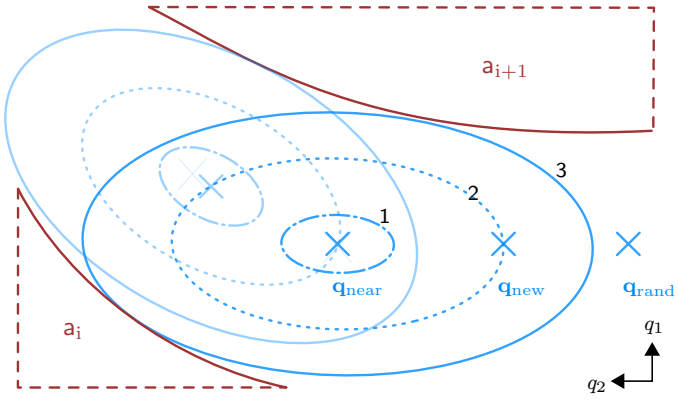

Fig. 1. By projecting a random joint space position q_{rand} to the hypersurface of $\mathbf{E}_{near}(\epsilon_1)$ of the nearest barrier pair, a new equilibrium of BP-RRT is created. The residue set $\mathbf{E}_{new}(\epsilon_0)$ of the new barrier pair is designed to be strictly inside the zero sub-level set $\mathbf{E}_{near}(1)$ of the nearest barrier pair. Notice that even if the undesirable regions of the workspace are polytopic, their joint space projections are not guaranteed to be also polytopic. The numbers indicate [1] the residue set $\mathbf{E}_{near}(\epsilon_0)$, [2] the hypersurface of $\mathbf{E}_{near}(\epsilon_1)$, and [3] the zero sub-level set $\mathbf{E}_{near}(1)$.

Similar to [11], the constraints Z_{safe}, Z_0 and U in (12), (13) and (14) can be transformed into LMIs

$$\begin{bmatrix} \bar{a}_i^2\mathbf{Q} & \star & \star & \star \\ \mathbf{0} & \gamma_i\mathbf{I} & \star & \star \\ \mathbf{a}_i\mathbf{J}_1\mathbf{S}_1\mathbf{Q} & \gamma_i\mathbf{a}_i\mathbf{J}_2 & 1 & \star \\ \mathbf{J}_3\mathbf{S}_1\mathbf{Q} & \mathbf{0} & \vec{0} & \gamma_i\mathbf{I} \end{bmatrix} \succeq 0,\ \forall i = 1,\dots,n_o \quad (17)$$

$$\begin{bmatrix} \bar{x}_i^2\mathbf{Q} & \star & \star & \star \\ \mathbf{0} & \mu_i\mathbf{I} & \star & \star \\ \mathbf{b}_i\mathbf{J}_1\mathbf{S}_1\mathbf{Q} & \mu_i\mathbf{b}_i\mathbf{J}_2 & 1 & \star \\ \mathbf{J}_3\mathbf{S}_1\mathbf{Q} & \mathbf{0} & \vec{0} & \mu_i\mathbf{I} \end{bmatrix} \succeq 0,\ \forall i = 1,\dots,n \quad (18)$$

$$\begin{bmatrix} \mathbf{Q} & \star \\ \mathbf{b}_i\mathbf{S}_2\mathbf{Q} & \bar{q}_i^2 \end{bmatrix} \succeq 0,\ \forall i = 1,\dots,n \quad (19)$$

$$\begin{bmatrix} \mathbf{Q} & \star \\ \mathbf{b}_i\mathbf{Y} & \bar{u}_i^2 \end{bmatrix} \succeq 0,\ \forall i = 1,\dots,n \quad (20)$$

where γ_i for $i = 1,\dots,n_o$ and μ_i for $i = 1,\dots,n$ are positive real scalar variables, $\mathbf{S}_2 \triangleq [\mathbf{0}_{n\times n}, \mathbf{I}_{n\times n}]$ and $\mathbf{Y} \triangleq \mathbf{K}\mathbf{Q}$.

By the following proposition, the robot's Lyapunov stability can be enforced under the impact of norm-bound human input $\mathbf{w}$.

Proposition 1: For a robot starting from a state in the zero sub-level set $\mathbf{E}(1)$ of the barrier pair $(\mathbf{B}, \mathbf{K})$, the robot state converges to a residue set $\mathbf{E}(\varepsilon_0)$ with an exponential convergence rate no less than $\frac{\alpha}{2}$ if

$$\begin{bmatrix} \mathbf{X}_{11} & \star \\ \mathbf{X}_{21} & \mathbf{X}_{22} \end{bmatrix} \preceq 0, \quad (21)$$

where

$$\mathbf{X}_{11} = \begin{bmatrix} \bar{\mathbf{A}}_1\mathbf{Q} + \mathbf{Q}\bar{\mathbf{A}}_1^\top + \bar{\mathbf{B}}_1^{\mathbf{u}}\mathbf{Y} + \mathbf{Y}^\top\bar{\mathbf{B}}_1^{\mathbf{u}\top} + \alpha\mathbf{Q} & \bar{\mathbf{B}}_1^{\mathbf{w}} \\ \bar{\mathbf{B}}_1^{\mathbf{w}\top} & -\alpha\frac{\varepsilon_0^2}{\bar{w}^2}\mathbf{I} \end{bmatrix}$$

$$+ \begin{bmatrix} \bar{\mathbf{A}}_2 & \bar{\mathbf{B}}_2^{\mathbf{u}} & \bar{\mathbf{B}}_2^{\mathbf{w}} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mu_x\mathbf{I} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mu_u\mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mu_w\mathbf{I} \end{bmatrix} \begin{bmatrix} \bar{\mathbf{A}}_2 & \bar{\mathbf{B}}_2^{\mathbf{u}} & \bar{\mathbf{B}}_2^{\mathbf{w}} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} \end{bmatrix}^\top \quad (22)$$

$$\mathbf{X}_{21} = \begin{bmatrix} \bar{\mathbf{A}}_3\mathbf{Q} & \mathbf{0} \\ \mathbf{B}_3^{\mathbf{u}}\mathbf{Y} & \mathbf{0} \\ \mathbf{0} & \mathbf{B}_3^{\mathbf{w}} \end{bmatrix} \quad (23)$$

$$\mathbf{X}_{22} = \begin{bmatrix} -\mu_x\mathbf{I} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & -\mu_u\mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & -\mu_w\mathbf{I} \end{bmatrix} \quad (24)$$

$$\bar{\mathbf{A}}_1 = \mathbf{S}_1^\top\mathbf{S}_2 + \mathbf{S}_2^\top\mathbf{A}_1\mathbf{S}_2,\ \bar{\mathbf{A}}_2 = \mathbf{S}_2^\top\mathbf{A}_2,\ \bar{\mathbf{A}}_3 = \mathbf{A}_3\mathbf{S}_2, \quad (25)$$

$$\bar{\mathbf{B}}_1^{\mathbf{u}} = \mathbf{S}_2^\top\mathbf{B}_1^{\mathbf{u}},\ \bar{\mathbf{B}}_2^{\mathbf{u}} = \mathbf{S}_2^\top\mathbf{B}_2^{\mathbf{u}}, \quad (26)$$

$$\bar{\mathbf{B}}_1^{\mathbf{w}} = \mathbf{S}_2^\top\mathbf{B}_1^{\mathbf{w}},\ \bar{\mathbf{B}}_2^{\mathbf{w}} = \mathbf{S}_2^\top\mathbf{B}_2^{\mathbf{w}}, \quad (27)$$

and μ_x, μ_u and μ_w are positive real scalar variables.

Proof: See Appendix A. ■

Finally, the volume of the ellipsoid $\mathbf{E}(1)$ is maximized through the cost function of the log of the determinant of $\mathbf{Q}$ [16]. A barrier pair synthesis sub-problem $(\mathbf{B}, \mathbf{k}) = \mathtt{BP}(\mathbf{x}_e, a_d, \bar{a}_1, \cdots, \bar{a}_{n_o}, Z_0, U, W)$ can be expressed as

$$\begin{aligned} \underset{\mathbf{Q},\mathbf{Y}}{\text{maximize}} \quad & \log(\det(\mathbf{Q})) \\ \text{subject to} \quad & \mathbf{Q} \succ 0, \\ & (16),\ (17),\ (18),\ (19),\ (20),\ (21) \end{aligned} \quad (28)$$

for finding a sub-level set $\mathbf{E}(1)$ that contains the desired region a_d and excludes the undesirable regions $a_1, a_2, \cdots, a_{n_o}$.

IV. Human-Robot Shared Autonomy

Based on the barrier pairs synthesized from the LMI problem formulated in (28), we propose a two-layer control framework for addressing the interaction problem defined in Section II.C. In the first layer, we employ the history of human input measurements to infer what the human wants the robot to do and define the robot's safety constraints based on it. In the second layer, we use the BP-RRT method [11] to generate sequences of barrier pairs that safely move the robot's end-effector to different desired regions. Based on the human intention inference performed in the first layer, the robot can execute these barrier pair sequences accordingly and help to accomplish the human's objective.

A. Barrier Pair Sampling

Although Algorithm 1 provides us the steps for creating a BP-RRT graph, it cannot be used directly to solve the interaction problem because of the additional human input $\mathbf{w}$. Therefore, we need to formulate a new barrier pair sampling algorithm for creating a barrier pair sequence that moves the robot's end-effector safely to a human desired region a_d under the human input intervention.

We extend our BP-RRT method to a robot under a human force input $\mathbf{w}$, as outlined in Algorithm 2. The algorithm initializes the BP-RRT graph by creating a barrier pair at a_f in line 1-3, expands it by sampling new barrier pairs in

Algorithm 2 $\mathbf{G} \leftarrow \texttt{BPRRT}(\mathsf{a}_0, \mathsf{a}_\mathsf{f}, \bar{\mathsf{a}}_1, \cdots, \bar{\mathsf{a}}_{\mathrm{n_o}}, \mathsf{Z}_0, \mathsf{U}, \mathsf{W}, \varepsilon_0, \varepsilon_1)$

Input: Initial region a_0, goal region a_f, constraints associated with undesirable regions $\bar{\mathsf{a}}_1, \cdots, \bar{\mathsf{a}}_{\mathrm{n_o}}$, state space constraint Z_0, robot input constraint U, human input constraint W, scalar ε_0 $(0 < \varepsilon_0 \leq 1)$, scalar ε_1 $(0 < \varepsilon_1 \leq 1)$

Output: BP-RRT graph $\mathbf{G}$

1: $(\mathbf{Q}_\mathsf{f}, \mathbf{K}_\mathsf{f}) \leftarrow \texttt{BP}(\mathbf{x}_{\mathsf{a}_\mathsf{f}}, \mathsf{a}_\mathsf{f}, \bar{\mathsf{a}}_1, \cdots, \bar{\mathsf{a}}_{\mathrm{n_o}}, \mathsf{Z}_0, \mathsf{U}, \mathsf{W}, \varepsilon_0)$

2: $\mathbf{G}.\texttt{AddVertex}(\mathbf{x}_\mathsf{f})$, $\mathbf{G}.\texttt{AddBP}((\mathbf{Q}_\mathsf{f}, \mathbf{K}_\mathsf{f}))$

3: $(\mathbf{Q}_\mathsf{new}, \mathbf{K}_\mathsf{new}) \leftarrow (\mathbf{Q}_\mathsf{f}, \mathbf{K}_\mathsf{f})$, $\mathbf{x}_\mathsf{new} \leftarrow \mathbf{x}_\mathsf{f}$

4: **while** $\mathbf{x}_0 \notin \mathbf{E}_\mathsf{new}(\varepsilon_1)$ **do**

5: $\quad \mathbf{q}_\mathsf{rand} \leftarrow \texttt{RandomConfiguration}(\bigcap_{\mathsf{i}=1}^{\mathrm{n_o}} \bar{\mathsf{a}}_\mathsf{i})$

6: $\quad \mathbf{E}_\mathsf{near}(\varepsilon_1) \leftarrow \texttt{NearestBP}(\mathbf{q}_\mathsf{rand}, \mathbf{G}, \varepsilon_1)$

7: $\quad \mathbf{q}_\mathsf{att} \leftarrow \texttt{NewEquilibrium}(\mathbf{q}_\mathsf{rand}, \mathbf{E}_\mathsf{near}(\varepsilon_1))$

8: $\quad \mathbf{x}_\mathsf{att} \leftarrow \mathbf{F}(\mathbf{q}_\mathsf{att})$

9: $\quad (\mathbf{Q}_\mathsf{att}, \mathbf{K}_\mathsf{att}) \leftarrow \texttt{BP}(\mathbf{x}_\mathsf{att}, \emptyset, \bar{\mathsf{a}}_1, \cdots, \bar{\mathsf{a}}_{\mathrm{n_o}}, \mathsf{Z}_0, \mathsf{U}, \mathsf{W}, \varepsilon_0)$

10: $\quad \varepsilon_2 \leftarrow \sqrt{[\mathbf{q}_\mathsf{near}^\top - \mathbf{q}_\mathsf{att}^\top, \vec{0}^\top]^\top \ \mathbf{Q}_\mathsf{att}^{-1} \ [\mathbf{q}_\mathsf{near}^\top - \mathbf{q}_\mathsf{att}^\top, \vec{0}^\top]}$

11: $\quad$ **if** $\mathbf{Q}_\mathsf{att} \preceq \frac{(1-\varepsilon_1)^2}{\varepsilon_0^2} \cdot \mathbf{Q}_\mathsf{near}$ **and** $\mathbf{Q}_\mathsf{near} \preceq \frac{(1-\varepsilon_2)^2}{\varepsilon_0^2} \cdot \mathbf{Q}_\mathsf{att}$ **then**

12: $\quad\quad (\mathbf{Q}_\mathsf{new}, \mathbf{K}_\mathsf{new}) \leftarrow (\mathbf{Q}_\mathsf{att}, \mathbf{K}_\mathsf{att})$, $\mathbf{x}_\mathsf{new} \leftarrow \mathbf{x}_\mathsf{att}$

13: $\quad\quad \mathbf{G}.\texttt{AddVertex}(\mathbf{x}_\mathsf{new})$, $\mathbf{G}.\texttt{AddBP}((\mathbf{Q}_\mathsf{new}, \mathbf{K}_\mathsf{new}))$, $\mathbf{G}.\texttt{AddEdge}((\mathbf{x}_\mathsf{near}, \mathbf{x}_\mathsf{new}))$

14: $\quad$ **end if**

15: **end while**

16: $(\mathbf{Q}_0, \mathbf{K}_0) \leftarrow \texttt{BP}(\mathbf{x}_{\mathsf{a}_0}, \mathsf{a}_0, \bar{\mathsf{a}}_1, \cdots, \bar{\mathsf{a}}_{\mathrm{n_o}}, \mathsf{Z}_0, \mathsf{U}, \mathsf{W}, \varepsilon_0)$

17: $\mathbf{G}.\texttt{AddVertex}(\mathbf{x}_0)$, $\mathbf{G}.\texttt{AddBP}((\mathbf{Q}_0, \mathbf{K}_0))$, $\mathbf{G}.\texttt{AddEdge}((\mathbf{x}_\mathsf{new}, \mathbf{x}_0))$

line 4-14, and completes it by creating a barrier pair at a_0 in line 15-16.

Different from Algorithm 1, Algorithm 2 considers two scalar inputs ε_0 and ε_1 (Fig. 1). The first scalar input ε_0, previously introduced in (22), defines the residue set $\mathbf{E}(\varepsilon_0)$ of a barrier pair. Similar to the scalar input ε in Algorithm 1, the second scalar input ε_1 defines a hyper-surface of sub-level set $\mathbf{E}_\mathsf{near}(\varepsilon_1)$ of the nearest barrier pair found in line 6 such that a new equilibrium $\mathbf{q}_\mathsf{att}$ can be obtained by projecting a random configuration $\mathbf{q}_\mathsf{rand}$ sampled in line 5 to this hyper-surface.

In order to enforce the robot's safe transition between two barrier pairs of an edge in the graph, the residue set $\mathbf{E}_\mathsf{att}(\varepsilon_0)$ of the newly sampled barrier pair created in line 9 needs to be completely inside the zero sub-level set $\mathbf{E}_\mathsf{near}(1)$ of the nearest barrier pair found in line 6. We can check this safety requirement through the condition stated in the following proposition.

Proposition 2: Suppose $(\mathbf{Q}_1, \mathbf{K}_1)$ and $(\mathbf{Q}_2, \mathbf{K}_2)$ represent two barrier pairs forming an edge in a BP-RRT graph. Let $\mathbf{z}_1 \triangleq [\mathbf{q}_1^\top, \vec{0}^\top]^\top$ and $\mathbf{z}_2 \triangleq [\mathbf{q}_2^\top, \vec{0}^\top]^\top$ be the equilibrium points of $(\mathbf{Q}_1, \mathbf{K}_1)$ and $(\mathbf{Q}_2, \mathbf{K}_2)$ located at the hyper-surface of $\mathbf{E}_2(\varepsilon_2)$ and $\mathbf{E}_1(\varepsilon_1)$, respectively. Let $\mathbf{E}_1(\varepsilon_0)$ and $\mathbf{E}_2(\varepsilon_0)$ be the residue sets of $(\mathbf{Q}_1, \mathbf{K}_1)$ and $(\mathbf{Q}_2, \mathbf{K}_2)$. The robot can safely transit between the zero sub-level sets $\mathbf{E}_1(1)$ and $\mathbf{E}_2(1)$ of these two barrier pairs if $\mathbf{Q}_1 \preceq \frac{(1-\varepsilon_2)^2}{\varepsilon_0^2} \cdot \mathbf{Q}_2$ and $\mathbf{Q}_2 \preceq \frac{(1-\varepsilon_1)^2}{\varepsilon_0^2} \cdot \mathbf{Q}_1$.

Proof: See Appendix B. ∎

Line 11 in Algorithm 2 checks the condition in Proposition 2. Notice that this condition guarantees the safe transition between $(\mathbf{Q}_\mathsf{near}, \mathbf{K}_\mathsf{near})$ and $(\mathbf{Q}_\mathsf{att}, \mathbf{K}_\mathsf{att})$ in both directions. Therefore, although the graph is initialized from a_f and expanded toward a_0, we can finally extract a sequence of barrier pairs which plan safe robot trajectories from a_0 to a_f and from a_f to a_0.

B. Human Intention Inference

Based on the concept of Boltzmann rationality [14], [15], we propose our human intent inference method for interpreting the human input $\mathbf{w}$ in the shared autonomy. Boltzmann rationality formalizes intent according to a variable that quantifies the value of the human's actions. In particular, it states that a rational human takes an action with probability proportional to the exponentiated value of that human action. Therefore, an action with higher value is more probable to be chosen by the human.

In the setting of robotic manipulation considered in this paper, we define the value of the human's action based on how well the human force $\mathbf{w}$ aligns with the direction toward the human's goal. Let a denote the human's goal, $\mathbf{x_t}$ denote the position of the robot's end-effector at time $\mathbf{t}$, and $\mathbf{w_t}$ denote the human force exerted at time $\mathbf{t}$. Recall that $\mathbf{x}_{\mathsf{a}_\mathsf{i}}$ is the center of a polytopic region a_i in the workspace. We define the likelihood function of exerting the force $\mathbf{w_t}$ conditioned on the true human's goal a as

$$\mathsf{p}(\mathbf{w_t} \mid \mathsf{a} = \mathsf{a}_\mathsf{i}) = \beta_0 \cdot \exp(\beta_1 \cdot \langle \mathbf{w_t}, \mathbf{x}_{\mathsf{a}_\mathsf{i}} - \mathbf{x_t} \rangle), \tag{29}$$

where $\beta_0 > 0$, $\beta_1 > 0$, and the value of the human's action is captured by the inner product of $\mathbf{w_t}$ and the direction toward the target region $\mathbf{x}_{\mathsf{a}_\mathsf{i}} - \mathbf{x_t}$. The value of this inner product indicates how well the exerted force is correlated with the direction toward the target region. β_0 is a partition function defined as

$$\beta_0^{-1} = \int_{\mathbf{w} \in \mathsf{W}} \exp(\beta_1 \cdot \langle \mathbf{w_t}, \mathbf{x}_{\mathsf{a}_\mathsf{i}} - \mathbf{x_t} \rangle) \, \mathrm{d}\mathbf{w}, \tag{30}$$

where W is the domain of feasible human force input defined in (15). β_1 is the rationality parameter representing the degree of human's rationality.

Now, using the likelihood function, we can compute and update the robot's belief over the human's intended target

region. Let us define the robot's belief as

$$\mathsf{b}_\mathrm{t}(\mathsf{a}_\mathrm{i}) = \mathsf{p}(\mathsf{a} = \mathsf{a}_\mathrm{i} \mid \mathbf{w}_0, \ldots, \mathbf{w_t}) \tag{31}$$

which denotes the probability of the target region being a_i given the history of human's inputs. Initially, the system starts with a uniform belief, i.e., $\mathsf{b}_0 \sim \mathtt{unif}\{1, \mathrm{n_o}\}$. Then, we can update the belief using the Bayes' theorem

$$\mathsf{b}_\mathrm{t}(\mathsf{a}_\mathrm{i}) = \frac{\mathsf{b}_{\mathrm{t}-1}(\mathsf{a}_\mathrm{i}) \cdot \mathsf{p}(\mathbf{w_t} \mid \mathsf{a} = \mathsf{a}_\mathrm{i})}{\sum_{\mathrm{j}=1}^{\mathrm{n_o}} \mathsf{b}_{\mathrm{t}-1}(\mathsf{a}_\mathrm{j}) \cdot \mathsf{p}(\mathbf{w_t} \mid \mathsf{a} = \mathsf{a}_\mathrm{j})}, \tag{32}$$

where $\mathsf{p}(\mathbf{w_t} \mid \mathsf{a} = \mathsf{a}_\mathrm{i})$ is computed according to (29).

At time $\mathbf{t}$, the robot's belief is used to select the sequence of barrier pairs that carry out the task of safely going to the estimated target region $\hat{\mathsf{a}}(\mathrm{t})$ calculated as

$$\hat{\mathsf{a}}(\mathrm{t}) = \underset{\mathrm{i} \in \{1,2,\ldots,\mathrm{n_o}\}}{\arg\max}\ \mathsf{b}_\mathrm{t}(\mathsf{a}_\mathrm{i}), \tag{33}$$

which has the highest probability of being the human's intended goal.

V. Example

In this section, we demonstrate the proposed control framework through a simulation of a 2-link manipulator robot with an equal length of $0.75\,\mathrm{m}$ for each link, a mass of $2.5\,\mathrm{kg}$ located at the distal end of each link, and a torque limit of $25\,\mathrm{N \cdot m}$ for each joint. Fig. 2 shows the polytopic regions in the workspace of the robot end effector, where a_1, a_2, and a_3 represent the desired task regions, a_4, a_5, and a_6 represent obstacle regions, and a_7 represents the region where the robot's base is located.

The manipulator robot starts from an end-effector position in a_1. A human operator decides whether to apply a 1 N force to the end-effector during the simulation. The human operator chooses the direction of the 1 N force from 8 different possible directions through a keyboard.

A. Barrier Pair Synthesis

We use Algorithm 2 to build barrier pair sequences which connect between a_1, a_2, and a_3 (Fig. 4.a-c). Barrier pairs c_1, c_2, and c_3 are in the middle of the sequences from a_2 to a_3, from a_3 to a_1, and from a_1 to a_2, respectively. Sometimes, the inference of the human target region may change and result in the barrier pair sequence currently executed by the robot to be invalid. Therefore, we also use Algorithm 2 to connect between c_1, c_2, and c_3 (Fig. 4.d-f) such that the robot can freely switch between the correct barrier pair sequences without going through any undesired regions. Fig. 3 shows the transitions between a_1, a_2, a_3, c_1, c_2, and c_3.

B. Simulation

The video of this simulation is available at `https://youtu.be/xTprU0jMT8w`.

The simulated manipulator robot uses the measurement of the 1 N force to infer the human operator's desired goal region. As the video shows, the initial force input from the human operator is sometimes ambiguous because it can point to multiple potential goal regions. However, the proposed human intention inference method is able to successfully recover the intended goal fast enough such that the manipulator robot does not move its end-effector to an incorrect goal region.

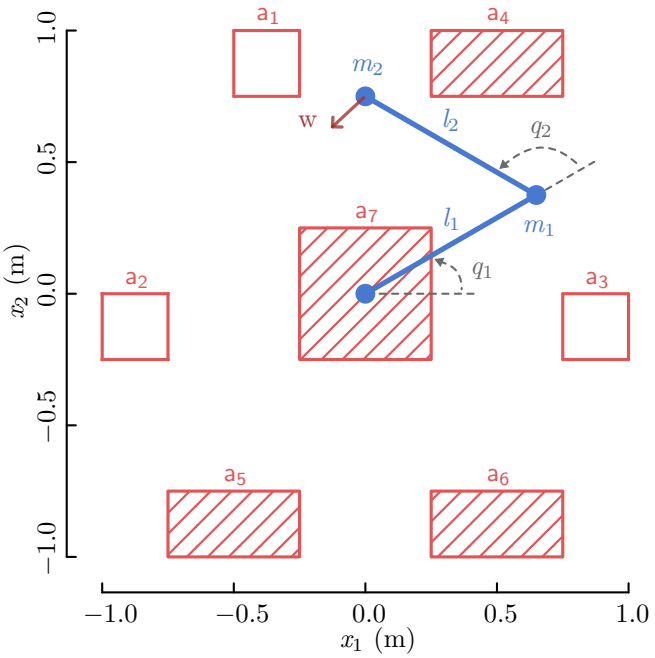

Fig. 2. A 2-link manipulator robot moves its end effector in a workspace with different polyoptic regions. a_1, a_2, and a_3 represent the desired task regions, a_4, a_5, and a_6 represent obstacle regions, and a_7 represents the region where the robot's base is located.

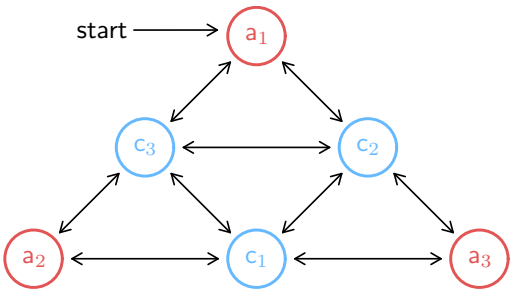

Fig. 3. A finite state machine indicates the possible transitions between a_1, a_2, a_3, c_1, c_2, and c_3.

VI. Discussion

In [11], the transition between two barrier pairs in a BP-RRT graph is unidirectional, so it is hard for the robot to return to its former location or switch between different barrier pair sequences. In this paper, we resolve this issue by enforcing the condition in Proposition 2. This condition is checked after the barrier pair synthesis because $\mathbf{Q}_\mathsf{near} \preceq \frac{(1-\varepsilon_2)^2}{\varepsilon_0^2} \cdot \mathbf{Q}_\mathsf{att}$ in line 11 of Algorithm 2 is a non-convex LMI constraint and cannot be included in the convex optimization problem defined in (28). However, this also means $\mathbf{Q}_\mathsf{att} \preceq \frac{(1-\varepsilon_1)^2}{\varepsilon_0^2} \cdot \mathbf{Q}_\mathsf{near}$ in line 11 of Algorithm 2 is a convex LMI constraint which can be potentially added to our barrier pair

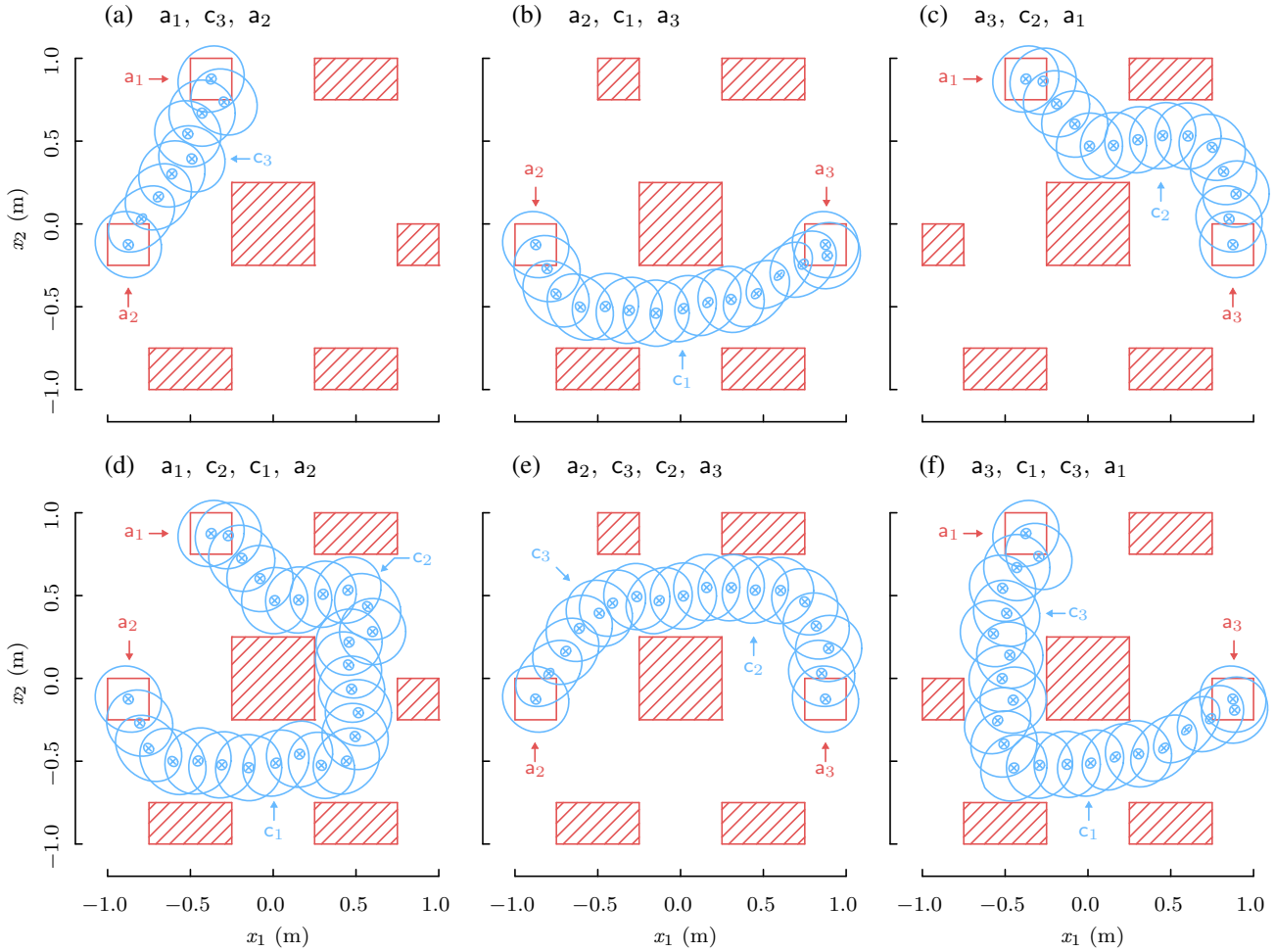

Fig. 4. Barrier pair sequences connect $\mathsf{a}_{\mathsf{init}}$ and $\mathsf{a}_{\mathsf{goal}}$ and avoid passing through the undesirable polytopic regions. The bigger and the smaller ellipsoids indicate the zero sub-level sets and the residue sets of the barrier pairs projected onto the x_1-x_2 space. The pointers indicate the locations of a_1, a_2, a_3, c_1, c_2, and c_3.

synthesis problem.

In the provided example, we create three additional barrier pair sequences (Fig. 4.d-f) to connect the midway barrier pairs c_1, c_2, and c_3 such that the robot's end-effector can safely switch between the three original barrier pair sequences (Fig. 4.a-c). But this strategy also creates more barrier pair sequences than necessary. The number of barrier pair sequences will increase dramatically if we consider more polytopic regions in Fig. 2 as the possible human desired regions. A potential approach to resolve this issue is considering all possible human desired regions in one barrier pair sampling algorithm and creating a roadmap [18] instead of just one sequence of barrier pairs.

APPENDIX

A. Proof of Proposition 1

Let us define $\mathbf{z} \triangleq [\tilde{\mathbf{q}}^\top, \dot{\tilde{\mathbf{q}}}^\top]^\top$. Based on the matrices defined in (25), (26) and (27), (10) can be expressed as

$$\dot{\mathbf{z}} = \bar{\mathbf{A}}_1\mathbf{z} + \bar{\mathbf{B}}_1^{\mathbf{u}}\mathbf{u} + \bar{\mathbf{B}}_1^{\mathbf{w}}\mathbf{w} + \bar{\mathbf{A}}_2\mathbf{p}_{\mathbf{z}} + \bar{\mathbf{B}}_2^{\mathbf{u}}\mathbf{p}_{\mathbf{u}} + \bar{\mathbf{B}}_2^{\mathbf{w}}\mathbf{p}_{\mathbf{w}} \quad (34)$$

where

$$\mathbf{p}_{\mathbf{z}} = \Delta\mathbf{q}_{\mathbf{z}}, \quad \mathbf{q}_{\mathbf{z}} = \bar{\mathbf{A}}_3\mathbf{z}, \quad (35)$$
$$\mathbf{p}_{\mathbf{u}} = \Delta\mathbf{q}_{\mathbf{u}}, \quad \mathbf{q}_{\mathbf{u}} = \mathbf{B}_3^{\mathbf{u}}\mathbf{u}, \quad (36)$$
$$\mathbf{p}_{\mathbf{w}} = \Delta\mathbf{q}_{\mathbf{w}}, \quad \mathbf{q}_{\mathbf{w}} = \mathbf{B}_3^{\mathbf{w}}\mathbf{w}. \quad (37)$$

For barrier function $\mathbf{B} = \mathbf{z}^\top\mathbf{P}\mathbf{z} - 1$ with controller $\mathbf{u} = \mathbf{K}\mathbf{z}$, the time derivative of $\mathbf{B}$ is

$$\dot{\mathbf{B}} = \begin{bmatrix} \mathbf{z} \\ \mathbf{w} \\ \mathbf{p_z} \\ \mathbf{p_u} \\ \mathbf{p_w} \end{bmatrix}^\top \begin{bmatrix} (\bar{\mathbf{A}}_1 + \bar{\mathbf{B}}_1^{\mathbf{u}}\mathbf{K})^\top\mathbf{P} + \mathbf{P}(\bar{\mathbf{A}}_1 + \bar{\mathbf{B}}_1^{\mathbf{u}}\mathbf{K}) & \star & \star & \star & \star \\ \bar{\mathbf{B}}_1^{\mathbf{w}\top}\mathbf{P} & 0 & \star & \star & \star \\ \bar{\mathbf{A}}_2^\top\mathbf{P} & 0 & 0 & \star & \star \\ \bar{\mathbf{B}}_2^{\mathbf{u}\top}\mathbf{P} & 0 & 0 & 0 & \star \\ \bar{\mathbf{B}}_2^{\mathbf{w}\top}\mathbf{P} & 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} \mathbf{z} \\ \mathbf{w} \\ \mathbf{p_z} \\ \mathbf{p_u} \\ \mathbf{p_w} \end{bmatrix}. \quad (38)$$

In addition, we have

$$\mathbf{w}^\top\mathbf{w} \leq \bar{w}^2, \quad (39)$$
$$\mathbf{z}^\top\mathbf{P}\mathbf{z} \leq \varepsilon_0^2, \quad (40)$$

for the norm-bound human input $\mathbf{w}$ and the residue set $\{\mathbf{z} \mid \mathbf{B} \leq \varepsilon_0^2 - 1\}$ of the barrier function.

Using the S-procedure, we can combine (35), (36), (37),

(39), (40), and $\dot{\mathbf{B}} \leq 0$ into

$$\begin{bmatrix} \tilde{\mathbf{X}}_{11} & \star & \star \\ \tilde{\mathbf{X}}_{21} & \tilde{\mathbf{X}}_{22} & \star \\ \mathbf{0} & \mathbf{0} & -\alpha\varepsilon_0^2 + \alpha_{\mathbf{w}}\bar{w}^2 \end{bmatrix} \preceq 0. \tag{41}$$

where

$$\tilde{\mathbf{X}}_{11} = \begin{bmatrix} (\bar{\mathbf{A}}_1 + \bar{\mathbf{B}}_1^{\mathbf{u}}\mathbf{K})^\top \mathbf{P} + \mathbf{P}(\bar{\mathbf{A}}_1 + \bar{\mathbf{B}}_1^{\mathbf{u}}\mathbf{K}) + \alpha\mathbf{P} & \mathbf{P}\bar{\mathbf{B}}_1^{\mathbf{w}} \\ \bar{\mathbf{B}}_1^{\mathbf{w}\top}\mathbf{P} & -\alpha_{\mathbf{w}}\mathbf{I} \end{bmatrix} + \begin{bmatrix} \bar{\mathbf{A}}_3 & \mathbf{0} \\ \mathbf{B}_3^{\mathbf{u}}\mathbf{K} & \mathbf{0} \\ \mathbf{0} & \mathbf{B}_3^{\mathbf{w}} \end{bmatrix}^\top \begin{bmatrix} \lambda_{\mathbf{x}}\mathbf{I} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \lambda_{\mathbf{u}}\mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \lambda_{\mathbf{w}}\mathbf{I} \end{bmatrix} \begin{bmatrix} \bar{\mathbf{A}}_3 & \mathbf{0} \\ \mathbf{B}_3^{\mathbf{u}}\mathbf{K} & \mathbf{0} \\ \mathbf{0} & \mathbf{B}_3^{\mathbf{w}} \end{bmatrix} \tag{42}$$

$$\tilde{\mathbf{X}}_{21} = \begin{bmatrix} \bar{\mathbf{A}}_2^\top \mathbf{P} & \mathbf{0} \\ \bar{\mathbf{B}}_2^{\mathbf{u}\top}\mathbf{P} & \mathbf{0} \\ \bar{\mathbf{B}}_2^{\mathbf{w}\top}\mathbf{P} & \mathbf{0} \end{bmatrix} \tag{43}$$

$$\tilde{\mathbf{X}}_{22} = \begin{bmatrix} -\lambda_{\mathbf{x}}\mathbf{I} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & -\lambda_{\mathbf{u}}\mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & -\lambda_{\mathbf{w}}\mathbf{I} \end{bmatrix} \tag{44}$$

Without loss of generality, we can let $\alpha_{\mathbf{w}} = \alpha\frac{\varepsilon_0^2}{\bar{w}^2}$ such that (41) becomes

$$\begin{bmatrix} \tilde{\mathbf{X}}_{11} & \star \\ \tilde{\mathbf{X}}_{21} & \tilde{\mathbf{X}}_{22} \end{bmatrix} \preceq 0 \tag{45}$$

which is equivalent to (21) for $\mu_{\mathbf{x}} = \frac{1}{\lambda_{\mathbf{x}}}$, $\mu_{\mathbf{u}} = \frac{1}{\lambda_{\mathbf{u}}}$, $\mu_{\mathbf{w}} = \frac{1}{\lambda_{\mathbf{w}}}$, and $\mathbf{Q} = \mathbf{P}^{-1}$.

B. Proof of Proposition 2

Let us can define a vector norm function

$$\|\star\|_{\mathbf{Q}_1} \triangleq \sqrt{\star^\top \mathbf{Q}_1^{-1} \star} \tag{46}$$

based on the quadratic part in the barrier function of $(\mathbf{Q}_1, \mathbf{K}_1)$. Because equilibrium $\mathbf{z}_2$ of $(\mathbf{Q}_2, \mathbf{K}_2)$ is on hyper-surface of $\mathbf{E}_1(\varepsilon_1)$, we have

$$\|\mathbf{z}_2 - \mathbf{z}_1\|_{\mathbf{Q}_1} = \varepsilon_1. \tag{47}$$

Suppose $\mathbf{z}_1'$ is a point on the hyper-surface of $\mathbf{E}_1(1)$, we have

$$\|\mathbf{z}_1' - \mathbf{z}_2\|_{\mathbf{Q}_1} + \|\mathbf{z}_2 - \mathbf{z}_1\|_{\mathbf{Q}_1} \geq \|\mathbf{z}_1' - \mathbf{z}_1\|_{\mathbf{Q}_1} = 1 \tag{48}$$

because of the triangle inequality of $\|\star\|_{\mathbf{Q}_1}$. Based on (47) and (48), we have

$$\|\mathbf{z}_1' - \mathbf{z}_2\|_{\mathbf{Q}_1} \geq 1 - \varepsilon_1 \tag{49}$$

which is equivalent to

$$\{\mathbf{z} \mid (\mathbf{z} - \mathbf{z}_2)^\top \mathbf{Q}_1^{-1}(\mathbf{z} - \mathbf{z}_2) \leq (1 - \varepsilon_1)^2\} \subseteq \mathbf{E}_1(1). \tag{50}$$

If $\mathbf{Q}_1 \preceq \frac{(1-\varepsilon_2)^2}{\varepsilon_0^2} \cdot \mathbf{Q}_2$, we have

$$\mathbf{E}_2(\varepsilon_0) \subseteq \{\mathbf{z} \mid (\mathbf{z} - \mathbf{z}_2)^\top \mathbf{Q}_1^{-1}(\mathbf{z} - \mathbf{z}_2) \leq (1 - \varepsilon_1)^2\}. \tag{51}$$

Combining (50) and (51), we have $\mathbf{E}_2(\varepsilon_0) \subseteq \mathbf{E}_1(1)$. If the robot starts from any states in $\mathbf{E}_2(1)$, it safely converges to a state in $\mathbf{E}_1(1)$ using barrier pair $(\mathbf{Q}_2, \mathbf{K}_2)$.

Similarly, we have $\mathbf{E}_1(\varepsilon_0) \subseteq \mathbf{E}_2(1)$ if $\mathbf{Q}_2 \preceq \frac{(1-\varepsilon_1)^2}{\varepsilon_0^2} \cdot \mathbf{Q}_1$.

REFERENCES

[1] E. Colgate, A. Bicchi, M. A. Peshkin, and J. E. Colgate, "Safety for physical human-robot interaction," in *Springer handbook of robotics*. Springer, 2008, pp. 1335–1348.

[2] T. Carlson and J. d. R. Millán, "Brain-controlled wheelchairs: A robotic architecture," *IEEE Robotics & Automation Magazine*, vol. 20, no. 1, pp. 65–73, 2013.

[3] L. Fridman, "Human-centered autonomous vehicle systems: Principles of effective shared autonomy," *arXiv preprint arXiv:1810.01835*, 2018.

[4] S. Javdani, H. Admoni, S. Pellegrinelli, S. S. Srinivasa, and J. A. Bagnell, "Shared autonomy via hindsight optimization for teleoperation and teaming," *The International Journal of Robotics Research*, vol. 37, no. 7, pp. 717–742, 2018.

[5] S. Prajna and A. Jadbabaie, "Safety verification of hybrid systems using barrier certificates," in *International Workshop on Hybrid Systems: Computation and Control*. Springer, 2004, pp. 477–492.

[6] K. P. Tee, S. S. Ge, and E. H. Tay, "Barrier Lyapunov functions for the control of output-constrained nonlinear systems," *Automatica*, vol. 45, no. 4, pp. 918–927, 2009.

[7] A. D. Ames, X. Xu, J. W. Grizzle, and P. Tabuada, "Control barrier function based quadratic programs for safety critical systems," *IEEE Transactions on Automatic Control*, vol. 62, no. 8, pp. 3861–3876, 2016.

[8] Q. Nguyen and K. Sreenath, "Optimal robust safety-critical control for dynamic robotics," *International Journal of Robotics Research (IJRR), in review*, 2016.

[9] D. Pylorof and E. Bakolas, "Analysis and synthesis of nonlinear controllers for input constrained systems using semidefinite programming optimization," in *American Control Conference (ACC)*. IEEE, 2016, pp. 6959–6964.

[10] G. C. Thomas, B. He, and L. Sentis, "Safety control synthesis with input limits: A hybrid approach," in *American Control Conference (ACC)*. IEEE, 2018, pp. 792–797.

[11] B. He, J. Lee, U. Topcu, and L. Sentis, "BP-RRT: Barrier pair synthesis for temporal logic motion planning," in *IEEE Conference on Decision and Control (CDC)*. IEEE, 2020, pp. 1404–1409.

[12] S. P. Buerger and N. Hogan, "Complementary stability and loop shaping for improved human–robot interaction," *IEEE Transactions on Robotics*, vol. 23, no. 2, pp. 232–244, 2007.

[13] B. He, G. C. Thomas, N. Paine, and L. Sentis, "Modeling and loop shaping of single-joint amplification exoskeleton with contact sensing and series elastic actuation," in *American Control Conference (ACC)*. IEEE, 2019, pp. 4580–4587.

[14] C. L. Baker, J. B. Tenenbaum, and R. R. Saxe, "Goal inference as inverse planning," in *Proceedings of the Annual Meeting of the Cognitive Science Society*, vol. 29, no. 29, 2007.

[15] O. Morgenstern and J. Von Neumann, *Theory of games and economic behavior*. Princeton university press, 1953.

[16] S. Boyd, L. El Ghaoui, E. Feron, and V. Balakrishnan, *Linear matrix inequalities in system and control theory*. SIAM, 1994, vol. 15.

[17] G. C. Thomas and L. Sentis, "Quadric inclusion programs: An LMI approach to H_∞-model identification,," *IEEE Transactions on Automatic Control*, vol. 64, no. 10, pp. 4229–4236, 2019.

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