A Hybrid Control Algorithm for Object Grasping Using Multiple Agents

Hyejin Han and Ricardo G. Sanfelice

***Abstract*— This paper presents a hybrid control approach for grasping objects by multiple agents without rebounding. When multiple agents grasp an object cooperatively, the motion of the agents is constrained due to the geometrical and frictional conditions at the contact points. In this paper, each agent acting on an object of interest is controlled by a hybrid controller which includes a position controller, a force controller, and some logic to coordinate grasping. The proposed approach provides a method to steer the agents to grasping positions on an object along appropriate directions and to asymptotically exert stabilizing forces at each contact point. The stability properties induced by the hybrid controller can be asserted using Lyapunov stability tools for hybrid systems. The set of allowed initial conditions guaranteed is characterized using sublevel sets of Lyapunov functions. The proposed algorithm is verified in simulations.**

I. Introduction

Despite decades of research, there still exists the challenge of performing robust autonomous grasping and manipulation by reliable physical interactions. Most grasping tasks require robust control of the contact forces and of the motion of the robot. Due to the fact that grasping tasks are accomplished through contacts, it is crucial to perform the transition between non-contact and contact motion without rebounding so that the contact between the robot and the grasped object is maintained once established.

Various approaches have been proposed to address the problem of controlling robots that execute tasks while in contact with the environment. In [1], a hybrid force and position controller was proposed to control both the motion of the robot and the force between the end-effector and its work environment. The solution in [1] provides a way to select compliant directions to interact with the environment. A dynamic hybrid control method, which takes the dynamics of the manipulator into consideration, is proposed in [2]. A unified approach for end-effector dynamic control and contact forces in the operational space framework has been proposed in [3]. In the operational space framework, end-effector tasks that involve constrained motions and contact forces are formulated. While most approaches have been developed for controlling the position and force simultaneously, these require the knowledge of the kinematics and dynamics of the entire robot [4], [5].

Switching strategies were proposed in [6]–[8] to overcome the problem of transition from unconstrained motion of the end-effector to constrained motion. However, it is often difficult to select appropriate gains and parameters of the controllers, and several bouncing events on the contact surface might happen before the contact is stable. Meanwhile, coordinated controls of multiple robots have been addressed in many scenarios, which include manipulation tasks such as grasping a large flexible object [9], [10]. Each robot has an independent controller and is autonomous, so that a decentralized framework can be applied to grasp an object.

In this paper, building from the control strategy in [11], a hybrid control algorithm for grasping tasks involving multiple agents is proposed. We provide a design procedure for the individual agents' hybrid controllers. Each hybrid controller is designed to control individual position and force, the latter relying only on contact force measurements to assure that rebounding does not occur from a certain compact set of initial conditions. With the proposed controller, each agent approaches the desired contact point along a constrained task direction and exerts the appropriate force onto the grasped object. The proposed algorithm guarantees that contact between all of the agents and the object occurs simultaneously and that the desired set points are robustly asymptotically stable.

This paper is organized as follows. Basic notions are summarized in Section II. Section III introduces the problem to be solved and outlines our proposed solution. The proposed solution is presented in more detail in Section V. The modeling, analysis, and design is performed in the framework of hybrid systems in Section III. Due to space constraints, some details and proofs are omitted and will be published elsewhere.

II. Preliminaries

A hybrid system $\mathcal{H} = (C, F, D, G)$ can be described as

$$\mathcal{H} \begin{cases} \dot{\mu} = F(\mu) & \mu \in C \\ \mu^+ = G(\mu) & \mu \in D \end{cases} \tag{1}$$

where $\mu \in \mathbb{R}^n$ is the state and $\mathbb{R}^n$ denotes n-dimensional Euclidean space, $F : \mathbb{R}^n \to \mathbb{R}^n$ denotes the flow map capturing the continuous dynamics on the flow set $C \subset \mathbb{R}^n$, and $G : \mathbb{R}^n \to \mathbb{R}^n$ defines the jump map capturing the discrete dynamics on the jump set $D \subset \mathbb{R}^n$. A solution to the hybrid system $\mathcal{H}$ is parametrized by $(t, j) \in \mathbb{R}_{\geq 0} \times \mathbb{N}$ where t is the ordinary time variable and j is the discrete jump variable, $\mathbb{R}$ denotes the real numbers, $\mathbb{R}_{\geq 0} := [0, \infty)$, and $\mathbb{N} := \{0, 1, 2, \ldots\}$. See [12] for more details about hybrid dynamical systems.

A function $\alpha : \mathbb{R}_{\geq 0} \times \mathbb{R}_{\geq 0} \mapsto \mathbb{R}_{\geq 0}$ is a class-$\mathcal{KL}$ function, also written $\beta \in \mathcal{KL}$, if it is nondecreasing in its first argument, nonincreasing in its second argument, $\lim_{r \to 0^+} \beta(r, s) = 0$ for each $s \in \mathbb{R}_{\geq 0}$, and $\lim_{s \to \infty} \beta(r, s) =$

This research has been partially supported by the National Science Foundation under CAREER Grant no. ECS-1450484, Grant no. ECS-1710621, and Grant no. CNS-1544396, by the Air Force Office of Scientific Research under Grant no. FA9550-16-1-0015, by the Air Force Research Laboratory under Grant no. FA9453-16-1-0053, and by CITRIS and the Banatao Institute at the University of California.

Hyejin Han and Ricardo G. Sanfelice are with the Computer Engineering Department, University of California, Santa Cruz, CA 95064, USA. hhan7,ricardo@ucsc.edu.

0 for each $r \in \mathbb{R}_{\geq 0}$. Given a set S and a point x, $\text{dist}(S, x)$ denotes the Euclidean distance from x to S.

III. Problem Statement and Outline of Hybrid Control Solution

We consider the problem of grasping an object at N contact points using the same number of end-effectors or agents evolving in space. Each contact point and associated force to exert, denoted $(x^d_{c,i}, y^d_{c,i}, z^d_{c,i}) \in \mathbb{R}^3$ and $f^d_{c,i} \in \mathbb{R}^3$ for each $i \in \mathcal{I} := \{1, 2, \ldots, N\}$, respectively, will serve as the reference to each of the agents; these are provided by a grasp generator. Motivated by the challenges outlined in Section I, the problem to solve is as follows:

Problem: Given an object to grasp, N agents to achieve contact with the object, and, for each $i \in \mathcal{I}$, contact positions $(x^d_{c,i}, y^d_{c,i}, z^d_{c,i})$ and desired contact forces $f^d_{c,i}$, design an algorithm to guarantee that the N agents establish simultaneous contact at points nearby $(x^d_{c,i}, y^d_{c,i}, z^d_{c,i})$ without rebounding and, after that, asymptotically exert force $f^d_{c,i}$ for each $i \in \mathcal{I}$. □

Our approach to providing a solution to this problem is to treat the grasping task involving multiple contact points as a multi-agent system in which each agent is commanded by an individual (hybrid) feedback controller, which, in turn, is coordinated by a (hybrid) supervisory algorithm. More precisely, a hybrid closed-loop system corresponding to agent i has state ξ_i and dynamics of the form

$$\mathcal{H}_i \begin{cases} \dot{\xi}_i = F_i(\xi_i) & \xi_i \in C_i \\ \xi_i^+ = G_i(\xi_i) & \xi_i \in D_i \end{cases} \quad i \in \mathcal{I}, \tag{2}$$

and the resulting system consists of N hybrid systems coordinated to perform the desired grasping task.

To solve the stated problem, we develop a hybrid controller supervising individual controllers in each agent, namely, position controllers and force controllers. Using the information provided by a grasp generator, the main task of each such controller is to first regulate position, so as to steer the agent to nearby the contact point simultaneously, and then regulate both position and force, so as to keep the vehicle nearby the contact point and exert the force needed to establish a stable grasp without rebounding. The supervisor employs the output of an algorithm providing the contact points and force needed to establish a stable grasp. Furthermore, when small perturbations are present in the system, which may trigger events of the supervisor at different time instances, contact between the agents and the object occur at times and at points that are nearby to those in the nominal conditions, and the resulting forces remain close to those determined by the grasp generator. The proposed logic does not incorporate avoidance strategies between the agents and the objects but that is part of future work.

IV. Object Grasping Using Multiple Agents

A. Agents model

We consider agents with dynamics in joint space given by

$$\mathcal{M}(\mathfrak{q})\ddot{\mathfrak{q}} + \mathcal{C}(\mathfrak{q}, \dot{\mathfrak{q}})\dot{\mathfrak{q}} + \mathcal{N}(\mathfrak{q}, \dot{\mathfrak{q}}) = \mathcal{T} - \mathcal{J}^\top(\mathfrak{q}) f_c \tag{3}$$

where $\mathcal{M}$ is the manipulator inertia matrix, $\mathcal{C}$ is the Coriolis matrix, $\mathcal{N}$ includes gravity terms and other forces that act at the joints, $\mathcal{T}$ is the vector of the actuators' torques, $\mathcal{J}$ is the Jacobian matrix relating the joint space velocity to the workspace velocity, and f_c is the vector of the contact forces due to the interaction between the manipulator and the environment. As the interest is in the interaction of N agents with an object, for each $i \in \mathcal{I}$, (3) is rewritten in the workspace coordinates $\chi_i = (x_i, y_i, z_i) \in \mathbb{R}^3$, after a coordinate transformation from $\mathfrak{q}$ to χ_i, which results in [13]

$$\widetilde{\mathcal{M}}_i \ddot{\chi}_i + \widetilde{\mathcal{C}}_i \dot{\chi}_i + \widetilde{\mathcal{N}}_i = F_i - f_{c,i} \tag{4}$$

where $\widetilde{\mathcal{M}}_i$, $\widetilde{\mathcal{C}}_i$, and $\widetilde{\mathcal{N}}_i$ (for simplicity their arguments are not included) are obtained from the matrices in joint space (namely, $\mathcal{M}_i$, $\mathcal{C}_i$, and $\mathcal{N}_i$; cf. (3)), and F_i is the vector of forces/torques applied at the end-effector of the i-th agent. Section V-B also illustrates the proposed hybrid controller on a nonholonomic model of the agents.

B. Object model

The surface of the object to grasp is assumed to be soft and so both the tangent plane and the normal can be defined at the contact points – these points are obtained from an algorithm that calculates a stable grasp; see Section IV-D. The object is defined by the set of points given by

$$\mathcal{W} = \{\chi : s(\chi) \leq 0\}$$

where s is a function that is smooth enough and χ are the workspace coordinates.

C. Contact force model

To characterize the relationship between the bodies' penetration and the reaction force involved in the end-effector/object interaction, we employ the so-called Kelvin-Voigt linear contact model. In such a model the viscoelastic material of the environment is described by the dynamics of a linear spring with stiffness k_c and damper coefficient b_c. The contact force $f_{c,i}$ is then given as follows [14]:

$$f_{c,i}(\chi_i, \dot{\chi}_i) = k_c \chi_{\ell,i} + b_c \dot{\chi}_{\ell,i} \tag{5}$$

when contact between the end-effector and the object occurs, and zero otherwise. The states $\chi_{\ell,i}$ and $\dot{\chi}_{\ell,i}$ are the compression distance and the compression velocity along the direction of contact, respectively – we use subindex ℓ to denote such local coordinates. Figure 1(a) depicts such local coordinates on the (x, y)-plane: red and green lines on the desired contact point refer to $x_{\ell,i}$-direction and $y_{\ell,i}$-direction, respectively.

D. Stable grasp generator

Given the object to grasp as defined in Section IV-B, we choose the contact points and forces in the workspace

$$(x^d_{c,i}, y^d_{c,i}, z^d_{c,i}) \in \mathbb{R}^3, \quad f^d_{c,i} \in \mathbb{R}^3 \qquad \forall i \in \mathcal{I} \tag{6}$$

that guarantee a stable grasp. The force balance equations for a grasped object subject to the contact force $f_{c,i}$ with a set of N contact points can be described as follows:

$$w = \widetilde{G} f \tag{7}$$

where w is the resulting wrench, $\widetilde{G} = [\widetilde{G}_1 \ \cdots \ \widetilde{G}_N]$ such that

$$\widetilde{G}_i = \begin{bmatrix} \overset{n_i}{} & \overset{s_i}{} & \overset{0}{} \\ (x^d_{c,i}, y^d_{c,i}, z^d_{c,i}) \times n_i & (x^d_{c,i}, y^d_{c,i}, z^d_{c,i}) \times s_i & n_i \end{bmatrix}, \tag{8}$$

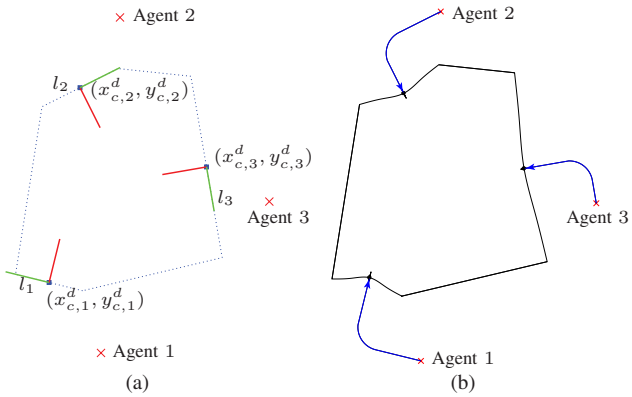


Fig. 1. Example of grasping task with three agents (i.e., $N=3$) on the (x,y)-plane. (a) For each agent i, the desired contact point $(x^d_{c,i}, y^d_{c,i})$ is given by the grasp generator. Red and green lines on the desired contact point refer to the $x_{\ell,i}$-direction and $y_{\ell,i}$-direction, respectively. (b) Position trajectory of each agent until the contact force is stabilized.

for each $i \in \mathcal{I}$; $f = (f^d_{c,1}, \ldots, f^d_{c,N})$, and n_i and s_i are the normal and tangent to the object at the i-th contact point, respectively. A solution (6) to (7) with $w=0$ defines a stable grasp, where $\widetilde{G}$ is determined by the contact force and object models.

There exist various approaches to optimize the placement of grasp points. In this paper, we use the method of maximizing grasp quality in [15]. Using the Ferrari-Canny metric, the most commonly used metric as a grasp quality evaluation, an optimal grasp minimizing the contact force magnitudes has been chosen among multiple different grasps satisfying (6) with $w=0$. We refer the reader to [16] for more details of the choice of the contact points and forces.

V. Hybrid Control Algorithm for Grasping using Multiple Agents

A. Hybrid Controller for Synchronized Grasping

The proposed controller is hybrid due to the combination of state variables that change continuously and, at times, jump [12], [17]. To coordinate each of the agents, the hybrid controller implements a supervisory logic that employs a controllable decreasing timer variable $\tau_i \in \mathbb{R}_{\geq 0}$ and a logic variable $q_i \in Q := \{-1, 0, 1, 2, 3\}$ for each $i \in \mathcal{I}$ agent. The timer state is used to schedule the steering of the agents so as to make contact with the object simultaneously. The five possible values of the logic variable represent the different modes of operation and phases therein – these are defined in the enumerated list below.

For each agent, the hybrid controller includes a position controller and a force controller for the purposes of controlling the position of the agents and the force exerted to the object. The position controller steers the agent to contact. The force controller employs measurements of the contact force in the direction of motion. To design these control algorithms, we follow the approaches in [3], [7], and [8]. First, we design the following inner feedback-linearizing loop that compensates for the internal and external forces of the manipulator, but certainly does not overcome the contact force:

$$F_i = u_i + \widetilde{\mathcal{C}}_i \dot{\chi}_i + \widetilde{\mathcal{N}}_i \tag{9}$$

where u_i is a new virtual control input. Then, as in [7], without loss of generality, we focus on the case in which the interaction between the agent and its environment occurs along a normal direction to the object, namely, the interaction between the agent and the object happens at a point on the line with that normal direction. We refer to this line as *the interaction line*; see Figure 1(a). Assuming that the mass is unitary, the dynamics of the agent along *the interaction line* is given as follows:

$$\ddot{\chi}_i = u_i - f_{c,i}(\chi_i, \dot{\chi}_i). \tag{10}$$

Using the contact force model in Section IV-C, once an agent reaches the surface of the object, the contact force is calculated based on the compression distance and velocity, respectively, for which, when focusing on the interaction between the agents and the object along the interaction line, results in $f_{c,i}$ in (5) being a scalar quantity given in local coordinates $x_{\ell,i}$ and $\dot{x}_{\ell,i}$; namely, when contact occurs,

$$f_{c,i}(\chi_i, \dot{\chi}_i) = k_c x_{\ell,i} + b_c \dot{x}_{\ell,i} \tag{11}$$

Note that the $x_{\ell,i}$-direction spans the interaction line as it is defined as the direction of the contact force.

Given the contact points and forces in (6) from the grasp generator, and assuming that the agents start far enough away from the object, the proposed supervisory logic for each agent $i \in \mathcal{I}$ is as follows:

1) The position controller initially steers the agents to (nearby) the interaction line. During this phase the i-th agent is in *position control mode*, for which $q_i = 0$.
2) When the agent position is close enough (characterized by the parameter $\varepsilon > 0$) to the interaction line, the agent enters into *waiting mode*, for which $q_i = 2$. In this mode the agent holds its position.
3) When all agents are in *waiting mode*, all logic variables are set to value 3 and the travel time of each agent to make contact with the object is computed by solving the closed-loop system dynamics. Then, the appropriate waiting duration for each agent is calculated by resetting τ_i to a nonnegative value chosen to guarantee that all agents establish contact with the object simultaneously when the timer expires. The decreasing timer counts down as long as it is nonnegative. The logic variable q_i of the agents in this phase remains at 3.
4) When τ_i reaches zero, the logic variable is reset to $q_i = 1$, and the agent enters another *position control mode* phase, but now the agent is directly steered towards the object along the interaction line.
5) When the contact force $f_{c,i}$ is larger than or equal to a certain threshold (denoted γ^i_2), we set $q_i = -1$ to put the agent in *force control mode*. The force controller is activated and the contact force is regulated to the magnitude of $f^d_{c,i}$. A switch back to the position controller is only possible when the contact force has decreased enough (characterized by a parameter γ^i_1, which is positive and strictly smaller than γ^i_2) – this hysteresis mechanism assures that rebounding does not occur and provides robustness to small perturbations; see [11] for more details.

As a result, all agents make contact with the object simultaneously and maintain contact at nearby the desired location without rebounds.

The logic outlined above can be modeled as a hybrid control algorithm given in terms of differential and difference equations with constraints. With such a model, the tools for stability analysis in [12], [17] are applied; see Section V-B. The state of the control algorithm is given by

$$\eta = (q_1, \tau_1, q_2, \tau_2, \ldots, q_N, \tau_N)$$

and its input is

$$u_c = ((\chi_1, \dot{\chi}_1), f_{c,1}, (\chi_2, \dot{\chi}_2), f_{c,2}, \ldots, (\chi_N, \dot{\chi}_N), f_{c,N})$$

plus the measurement noise signal on $(\chi_i, \dot{\chi}_i)$ and $f_{c,i}$

$$m = (m_1, m_{f_{c,1}}, m_2, m_{f_{c,2}}, \ldots, m_N, m_{f_{c,N}}).$$

Next, we provide the differential and difference equations, along with the constraints, for each agent $i \in \mathcal{I}$ modeling the proposed control algorithm. Below, l_i denotes the i-th interaction line.

Flows: The continuous change of the logic variable q_i is given by the (trivial) differential equation

$$\dot{q}_i = 0 \tag{12}$$

which always keeps the logic variables constant in the continuous-time regime. The timer variable τ_i continuously decrements itself according to the differential equation

$$\dot{\tau}_i = -1 \tag{13}$$

when $q_i = 3$ and $\tau_i \geq 0$ and, for any other values of q_i and τ_i, τ_i changes (trivially) according to

$$\dot{\tau}_i = 0 \tag{14}$$

Jumps: The jumps of the hybrid controller update the variables q_i and τ_i so as to implement the logic above. These updates are instantaneous and governed by the following difference equations with constraints:

a) From *position control mode* to *waiting mode*:

$$q_i^+ = 2, \qquad \tau_i^+ = \tau_i \tag{15}$$

when $q_i = 0$ and $\text{dist}(l_i, \chi_i) \leq \varepsilon$

b) From *waiting mode* to *position control mode*:

b1) to steering toward the object (first phase):

$$q_i^+ = 1, \qquad \tau_i^+ = \tau_i \tag{16a}$$

when $q_i = 3$, $\text{dist}(l_i, \chi_i) \leq \varepsilon$ and $\tau_i \leq 0$,

b2) to steering back to nearby the line (second phase):

$$q_i^+ = 0, \qquad \tau_i^+ = \tau_i \tag{16b}$$

when $q_i \in \{2, 3\}$ and $\text{dist}(l_i, \chi_i) \geq \varepsilon'$, $\varepsilon' > \varepsilon$

c) From *position control mode* to *force control mode*:

$$q_i^+ = -1, \qquad \tau_i^+ = \tau_i \tag{17}$$

when $q_i = 1$ and $f_{c,i} \geq \gamma_2^i$

d) From *force control mode* to *position control mode*:

$$q_i^+ = 1, \qquad \tau_i^+ = \tau_i \tag{18}$$

when $q_i = -1$ and $f_{c,i} \leq \gamma_1^i$

e) From *waiting mode* (first phase) to *waiting mode* (second phase) when all agents are nearby their respective interaction line:

$$q_i^+ = 3, \qquad \tau_i^+ = T_i(\chi_i) \tag{19}$$

when $q_i = 2$ for all $i \in \mathcal{I}$, and T_i is the waiting time that is calculated based on the time to reach the surface of the object for each agent i.

The output of the hybrid controller assigns the virtual input u_i (see (9)) of the agents as follows:

$$u_i = \begin{cases} \kappa_P(\chi_i, \dot{\chi}_i) & \text{if } q_i \in \{0, 1\} \\ \kappa_F(\chi_i, \dot{\chi}_i, f_{c,i}) & \text{if } q_i = -1 \\ (0, 0) & \text{if } q_i \in \{2, 3\} \end{cases} \tag{20}$$

where κ_P is the position controller and κ_F is the force controller. Note that when the agent is in waiting mode, its input is identically zero so as to wait at the current location. The time to reach the surface of the object, namely, T_i, can be analytically computed once the position controller is designed. More details on how to design such feedback laws using Lyapunov theory are provided next.

B. Main Results

The state of each individual agent is denoted $\eta_i \in \mathbb{R}^3$ and represents position and velocity in the local frame.[1] For agent $i \in \mathcal{I}$, the states $\eta_{i,1}$ and $\eta_{i,2}$ are the position and velocity in the $x_{\ell,i}$-direction, and $\eta_{i,3}$ is the position in the $y_{\ell,i}$-direction. As stated in Section V-A, the i-th hybrid controller employs τ_i and q_i to implement a supervisory logic. The logic therein leads to an i-th hybrid closed-loop system $\mathcal{H}_i = (C_i, F_i, D_i, G_i)$ as in (2) with state $\xi_i := (\eta_i, \tau_i, q_i) \in Z := \mathbb{R}^3 \times \mathbb{R}_{\geq 0} \times Q$. The controller parameters k_p^i, k_d^i are the proportional and derivative feedback gains of the $x_{\ell,i}$ position controller, respectively, and $k_{p,y}^i$ is the proportional feedback gain of the $y_{\ell,i}$ position controller. The parameter k_f^i is the proportional feedback gain of the force controller.

For each $i \in \mathcal{I}$, we define $\mathcal{A}_i := (x_{\ell,i}^F, 0, y_{\ell,i}^d)$ where $x_{\ell,i}^F = f_{c,i}^d / k_c$. Then, given parameters $k_c, b_c \in (0, +\infty)$ of the work environment and desired contact force $0 < f_{c,i}^d < \hat{f}_{c,i}$ where $\hat{f}_{c,i}$ is the maximum allowed force, one can always find

1) compact sets $K_{0,i}, K_{1,i}, K_{2,i} \subset \mathbb{R}^3$,
2) parameters $k_p^i, k_d^i, k_{p,y}^i, k_f^i, \gamma_1^i, \gamma_2^i, x_{\ell,i}^d$ of the hybrid controller

such that the set $\mathcal{A}_i \times \{0\} \times \{-1\}$ is locally asymptotically stable with basin of attraction containing $((K_{0,i} \times \{0\} \times \{1\}) \cup (K_{1,i} \times \{0\} \times \{0\}) \cup (K_{2,i} \times \{0\} \times \{-1\}))$ for $\mathcal{H}_i$. In fact, a particular choice of these sets is

- $K_{0,i} = (L_{V_1}(r_1) \cap \{\eta_i \in \mathbb{R}^3 : \eta_{i,1} \leq 0\}) \cup (L_{V_2}(r_2) \cap \{\eta_i \in \mathbb{R}^3 : \eta_{i,1} \geq 0\})$ where $x_{\ell,i}^d, k_p^i, k_d^i > 0$, $V_1(\eta_{i,1}, \eta_{i,2}) = \frac{1}{2} a_1 (\eta_{i,1} - x_{\ell,i}^d)^2 + \frac{1}{2} b_1 \eta_{i,2}^2$ with a_1, b_1 satisfying $\frac{a_1}{b_1} = k_p^i$, and $V_2(\eta_{i,1}, \eta_{i,2}) = \frac{1}{2} a_2 (\eta_{i,1} - x_{\ell,i}^P)^2 + \frac{1}{2} b_2 \eta_{i,2}^2$ with a_2, b_2 satisfying $\frac{a_2}{b_2} = k_p^i + k_c$ where $x_{\ell,i}^P := \frac{k_p^i}{k_p^i + k_c} x_{\ell,i}^d$; r_1 and r_2 are the maximum value of the level set of V_1 and V_2, respectively, such that the following conditions are satisfied: the r_1-level set of V_1 intersects the point $\eta_{i,1} = 0$, $\eta_{i,2} = \eta_{i,2}^* - \delta$ and $r_2 = \min\{r_2^a, r_2^b\}$, where the r_2^a-level set of V_2 is such that it crosses the intersection of the r_F-level set and the min γ_2^i line, and the r_2^b-level set of V_2 is such that it intersects the point $\eta_{i,1} = 0$, $\eta_{i,2} = \eta_{i,2}^* - \delta$, where $\eta_{i,2}^*$ is the maximum value of the velocity allowed at the contact location and $\delta \in (0, \eta_{i,2}^*)$;

[1] For simplicity, we write it in local coordinates. The global coordinates case requires replacing η_i by $\Phi(\eta_i)$ where $\Phi(\eta_i)$ is a transformation involving both rotation and translation of the coordinates.

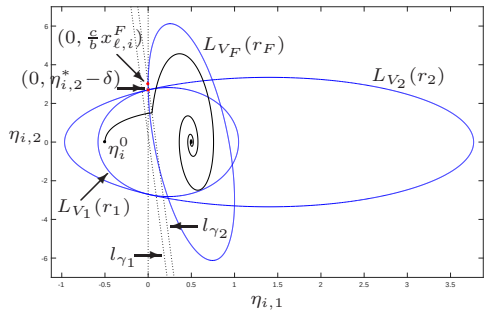


Fig. 2. Example of sublevel sets of Lyapunov functions. $\eta^0_i := (\eta^0_{i,1}, \eta^0_{i,2})$ is the initial point. The lines l_{γ_1} and l_{γ_2} are given by $l_{\gamma_1} := \{(\eta_{i,1}, \eta_{i,2}) : \eta_{i,2} = -\frac{k_c}{b_c}\eta_{i,1} + \frac{\gamma_1}{b_c}\}$ and $l_{\gamma_2} := \{(\eta_{i,1}, \eta_{i,2}) : \eta_{i,2} = -\frac{k_c}{b_c}\eta_{i,1} + \frac{\gamma_2}{b_c}\}$.

- $K_{1,i} = L_{V_3}(r_3)$ where $V_3(\eta_{i,3}) = \frac{1}{2}a_3(\eta_{i,3} - y^d_{\ell,i})^2, a_3 > 0, k^i_{p,y} > 0$, and $r_3 > 0$;
- $K_{2,i} = L_{V_F}(r_F)$ where $V_F(\eta_{i,1}, \eta_{i,2}) := a(\eta_{i,1} - x^F_{\ell,i})^2 + b\eta^2_{i,2} + 2c(\eta_{i,1} - x^F_{\ell,i})\eta_{i,2}$, and $P_F := \begin{bmatrix} a & c \\ c & d \end{bmatrix} = R\begin{bmatrix} p_1 & 0 \\ 0 & p_2 \end{bmatrix}R^\top$, $p_1, p_2 > 0, R := \begin{bmatrix} -\sin\beta & -\cos\beta \\ \cos\beta & -\sin\beta \end{bmatrix}$, and $\beta := \arctan(-k_c/b_c)$, $k^i_f \in \left(0, \frac{-2c^2k_c + abk_c + acb_c}{(bk_c - cb_c)^2}\right)$; r_F is the value of the level set of V_F when V_F is at $\eta_{i,1} = 0$, $\eta_{i,2} = \frac{c}{b}x^F_{\ell,i}$ where $x^F_{\ell,i} := f^d_{c,i}/k_c$; and

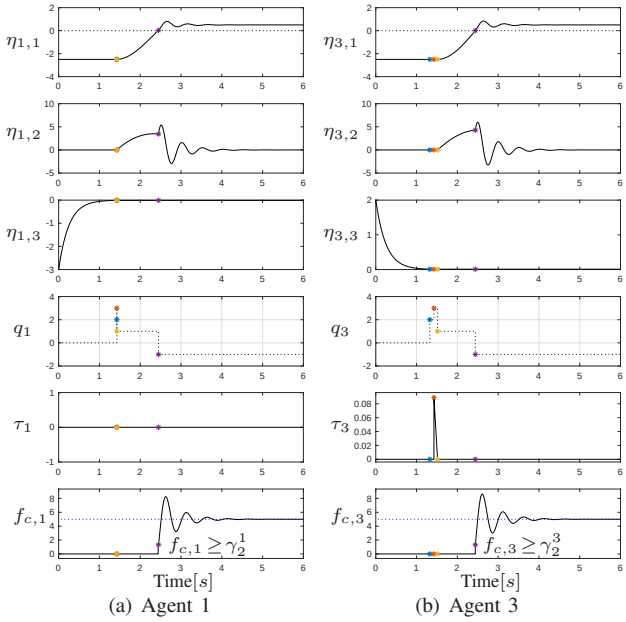
the parameters in 2) can be chosen as follows: $\gamma^i_{1,\min} = 0$, $\gamma^i_{1,\max} = x^F_{\ell,i}\left(k_c - \sqrt{\frac{k_c^2 b - 2ck_cb_c + ab_c^2}{b}}\right)$, $\gamma^i_{2,\min} = b_c\frac{c}{b}x^F_{\ell,i}$, $\gamma^i_{2,\max} = k_c \min\{\frac{k_p}{k_p + k_c}x^d_{\ell,i}, x^F_{\ell,i}\}$, $x^d_{\ell,i} \in [x^d_{\ell,i_{\min}}, +\infty]$ where $x^d_{\ell,i_{\min}} = x^F_{\ell,i}b_c\frac{c}{b}\frac{k_p + k_c}{k_pk_c}$. Since the hybrid closed-loop system satisfies the hybrid basic conditions (see [17]), we can find $\beta \in \mathcal{KL}$ such that for each $\epsilon > 0$ and each compact set $K_{i,0}, K_{i,1}, K_{i,2} \subset \mathbb{R}^3$ such that $((K_{0,i} \times \{0\} \times \{1\}) \cup (K_{1,i} \times \{0\} \times \{0\}) \cup (K_{2,i} \times \{0\} \times \{-1\}))$ is a subset of the basin of attraction of $\mathcal{H}_i$, there exists $\delta^* > 0$ such that for each position and force measurement noise $m : \mathbb{R}_{\geq 0} \to \delta^*\mathbb{B}$, solutions ξ_i to $\mathcal{H}_i$ with noise m for initial conditions $z^0_i \in ((K_{0,i} \times \{0\} \times \{1\}) \cup (K_{1,i} \times \{0\} \times \{0\}) \cup (K_{2,i} \times \{0\} \times \{-1\}))$ are such that the η_i component of the solutions satisfy $|\eta_i(t,j)|_{\mathcal{A}} \leq \beta(|\eta^0_i|_{\mathcal{A}}, t + j) + \epsilon$ for each $(t,j) \in \operatorname{dom}\xi_i$. Figure 2 shows an example of sublevel sets of Lyapunov functions above.

C. Nominal case

Now, we illustrate the design conditions above. We consider the task of grasping an object with multiple contact points. The proposed hybrid system is simulated with MATLAB using the Hybrid Equation (HyEQ) Toolbox [18]. The simulation results show how the proposed controller stabilizes the horizontal and vertical positions and ensures contact force regulation in the multi-agent systems.[2]

In the following simulation, we apply the proposed hybrid controller in Section V-A for $N = 3$ to grasp an object defined on the (x, y)-plane as a polygon with vertices (in clockwise order) given by $\{(-1.89, -4.95), (-4.99, -4.22), (-3.70, 3.26), (-0.31, 4.96), (2.94, 4.62), (4.34, -3.48)\}$; see Figure 1. The parameters k_c and b_c are set to 10 and 0.3, respectively. For each $i \in \mathcal{I} := \{1, 2, 3\}$, the gains k^i_f, k^i_p and $k^i_{p,y}$ are set to 16.0, 2.0 and 4.0, respectively; the set of gains (k^1_d, k^2_d, k^3_d) is set to $(1, 0.5, 0.5)$. The desired contact forces $f^d_{c,i}$ obtained from the stable grasp generator are 4.39, 4.2 and 2.97, respectively, at each contact point. The thresholds γ^i_1 and γ^i_2 are chosen as 0.76 and 1.33, respectively. The thresholds ε and ε' are set to 0.01 and 0.05, respectively. The initial condition of each agent is $(\eta^0_{1,1}, \eta^0_{1,2}, \eta^0_{1,3}) = (-0.5, 0, 0)$, $(\eta^0_{2,1}, \eta^0_{2,2}, \eta^0_{2,3}) = (-0.5, 0, 1)$, $(\eta^0_{3,1}, \eta^0_{3,2}, \eta^0_{3,3}) = (1, 0, 0)$, respectively.

[2]Simulation code can be found at https://github.com/HybridSystemsLab/MultiAgentGrasping



(a) Agent 1 (b) Agent 3

Fig. 3. Grasping task with three agents: Plots of state variables of agent 1 and agent 3 in their local frames.

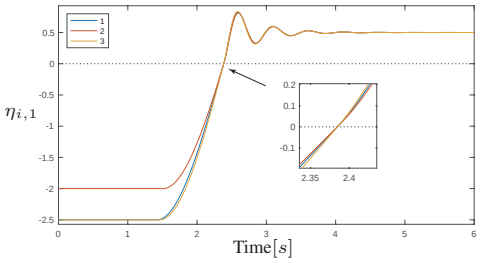


Fig. 4. Plots of $\eta_{i,1}$ corresponding to each agent.

Figure 3 illustrates the closed-loop trajectory of the three agents obtained from the simulation. The vertical position controller is initially applied (i.e., $q_i = 0$). Approximately 1.5 seconds later, when $\operatorname{dist}(l_i, \xi_i) = |\eta_{i,3} - y^d_{\ell,i}| \leq \varepsilon$, the horizontal position controller is applied (i.e., $q_i = 1$). When the contact force $f_{c,i} \geq \gamma^i_2$, the force controller is activated (i.e., $q_i = -1$). The contact force is regulated to $f^d_{c,i}$ by using the force controller. As the plots of the contact forces in Figure 3 indicate, the proposed hybrid controller guarantees that the agents do not bounce off the surface of the object after contact.

As shown in Figure 4, at approximately 2.5 seconds, all three agents make contact with the object (i.e., $\{x_{i,1}\}_{i=1}^3 = 0$) simultaneously. A movie of this simulation is available at

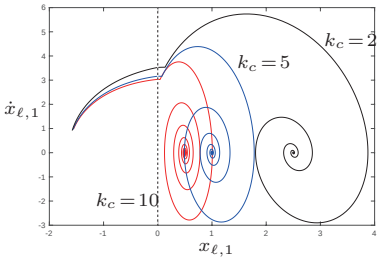
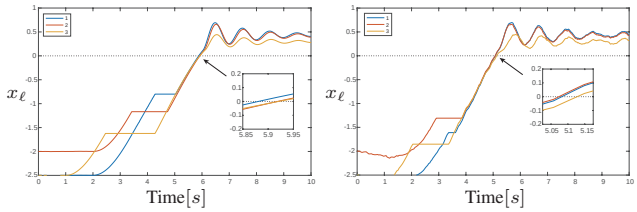


Fig. 5. Position vs. velocity plots for agent 1 corresponding to different environment material stiffness k_c. The parameter k_c is set to 2, 5 and 10, respectively. Agent 1 does not bounce off the surface of the object after contact (i.e., $x_{\ell,1} = 0$).



(a) $\sigma = 0.01$ (b) $\sigma = 0.25$

Fig. 6. Noise case: Plots of local position $x_{\ell,i}$ for each agent.

`https://youtu.be/6B8m584u-g4`.

Figure 5 shows the trajectories with respect to the local coordinates for different environments. After each agent makes contact with the object (i.e., $x_{\ell,i} = 0$), it maintains its contact with the object while the position of each agent is stabilized. Note that the different environment material stiffness k_c, the equilibrium point $x^F_{\ell,i}$ is changed according to $x^F_{\ell,i} = |f^d_{c,i}|/k_c$ so as to exert the desired contact force magnitudes of $f^d_{c,i}$.

D. Noise case of agents having Dubins-like dynamics

Figure 6 shows the trajectories of the position $x_{\ell,i}$ corresponding to each agent with different values of noise. In this section, motivated by the wide applicability of Dubins-type models, we apply our hybrid controller to the car-type model given by

$$\begin{aligned} \dot{x}_{\ell,i} &= v_{\ell,i}\cos\theta_{\ell,i}, \quad \dot{y}_{\ell,i} = v_{\ell,i}\sin\theta_{\ell,i}, \\ \dot{v}_{\ell,i} &= u_{v,i} - f_{c,i}, \quad \dot{\theta}_{\ell,i} = u_{\theta,i} \end{aligned} \tag{21}$$

where, for each $i \in \mathcal{I}$ agent, $(x_{\ell,i}, y_{\ell,i}) \in \mathbb{R}^2$ denotes planar position, $\theta_{\ell,i} \in \mathbb{R}$ denotes orientation and $v_{\ell,i} \in \mathbb{R}$ denotes the forward velocity, respectively; see, e.g., [19]. The inputs $u_{\theta,i}$ and $u_{v,i}$ are the angular velocity input and the acceleration input, respectively. The norm of the angular velocity input is upper bounded by the constant $\overline{u}_{\theta,i}$, which implies that the vehicle turns have a (nonzero) minimum turning radius. In other words, given an input signal $(u_{\theta,i}, u_{v,i})$, the resulting paths in the (x, y)-plane have bounded curvature.

In the following simulation, the initial condition of each agent $i \in \mathcal{I} := \{1, 2, 3\}$ is $(x^0_{\ell,1}, y^0_{\ell,1}, v^0_{\ell,1}, \theta^0_{\ell,1}) = (-2.5, -3, 1, \frac{\pi}{2})$, $(x^0_{\ell,2}, y^0_{\ell,2}, v^0_{\ell,2}, \theta^0_{\ell,2}) = (-2, 3, 1, -\frac{\pi}{2})$ and $(x^0_{\ell,3}, y^0_{\ell,3}, v^0_{\ell,3}, \theta^0_{\ell,3}) = (-2.5, 2, 1, -\frac{\pi}{2})$, respectively. A Gaussian noise with zero mean and variance of $\sigma = 0.01, 0.03, 0.05, 0.1, 0.2, 0.25$, respectively, define the noise signals $f_{c,i}$ and m_i for each $i \in \mathcal{I}$. The contact time t^i_c is changed under different noises, but mismatch of contact time is approximately 0.08 seconds in the worst case (Table I).

Compared to the nominal case, mismatch of the equilibrium points is approximately 0.1 (in norm) in the worst case. Please go to `https://youtu.be/7VPnZj6a7Bo` to watch a movie of this simulation.

σ	$\lvert t^1_c - t^2_c\rvert$	$\lvert t^1_c - t^3_c\rvert$	$\lvert t^2_c - t^3_c\rvert$
0	0	0	0
0.01	0.05	0.05	0
0.03	0.05	0.08	0.05
0.05	0.05	0.05	0
0.1	0	0.08	0.08
0.2	0.05	0.025	0.025
0.25	0.01	0.05	0.05

TABLE I

CONTACT TIME OF THREE AGENTS.

REFERENCES

[1] M. H. Raibert and J. J. Craig, "Hybrid position/force control of manipulators," *Journal of Dynamic Systems, Measurement, and Control*, vol. 103, no. 2, pp. 126–133, 1981.

[2] T. Yoshikawa, "Dynamic hybrid position/force control of robot manipulators–description of hand constraints and calculation of joint driving force," *IEEE Journal on Robotics and Automation*, vol. 3, no. 5, pp. 386–392, 1987.

[3] O. Khatib, "A unified approach for motion and force control of robot manipulators: The operational space formulation," *IEEE Journal on Robotics and Automation*, vol. 3, no. 1, pp. 43–53, 1987.

[4] B. Siciliano and L. Villani, "A passivity-based approach to force regulation and motion control of robot manipulators," *Automatica*, vol. 32, no. 3, pp. 443–447, 1996.

[5] R. Featherstone, S. Sonck, and O. Khatib, "A general contact model for dynamically-decoupled force/motion control," in *Experimental Robotics V*. Springer, 1998, pp. 128–139.

[6] J. K. Mills and D. M. Lokhorst, "Stability and control of robotic manipulators during contact/noncontact task transition," *IEEE Transactions on Robotics and Automation*, vol. 9, no. 3, pp. 335–345, 1993.

[7] T.-J. Tarn, Y. Wu, N. Xi, and A. Isidori, "Force regulation and contact transition control," *IEEE Control Systems*, vol. 16, no. 1.

[8] M. C. Cavusoglu, J. Yan, and S. S. Sastry, "A hybrid system approach to contact stability and force control in robotic manipulators," in *Proceedings of 12th IEEE International Symposium on Intelligent Control*, 1997, pp. 143–148.

[9] T. G. Sugar and V. Kumar, "Control of cooperating mobile manipulators," *IEEE Transactions on robotics and automation*, vol. 18, no. 1, pp. 94–103, 2002.

[10] Z. Li, J. Li, and Y. Kang, "Adaptive robust coordinated control of multiple mobile manipulators interacting with rigid environments," *Automatica*, vol. 46, no. 12, pp. 2028–2034, 2010.

[11] R. Carloni, R. G. Sanfelice, A. R. Teel, and C. Melchiorri, "A hybrid control strategy for robust contact detection and force regulation," in *Proceedings of the American Control Conference*, 2007.

[12] R. Goebel, R. G. Sanfelice, and A. R. Teel, "Hybrid dynamical systems," *IEEE Control Systems*, vol. 29, no. 2, pp. 28–93, 2009.

[13] J. Park and O. Khatib, "Robot multiple contact control," *Robotica*, vol. 26, no. 5, pp. 667–677, 2008.

[14] G. Gilardi and I. Sharf, "Literature survey of contact dynamics modelling," *Mechanism and machine theory*, vol. 37, no. 10, 2002.

[15] S. Liu and S. Carpin, "Fast grasp quality evaluation with partial convex hull computation," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2015, pp. 4279–4285.

[16] S. Carpin, S. Liu, J. Falco, and K. Van Wyk, "Multi-fingered robotic grasping: A primer," *arXiv preprint arXiv:1607.06620*, 2016.

[17] R. Goebel, R. G. Sanfelice, and A. R. Teel, *Hybrid Dynamical Systems: Modeling, Stability, and Robustness*. New Jersey: Princeton University Press, 2012.

[18] R. G. Sanfelice, D. Copp, and P. Nanez, "A toolbox for simulation of hybrid systems in matlab/simulink: Hybrid equations (hyeq) toolbox," in *Proceedings of the 16th international conference on Hybrid systems: computation and control*. ACM, 2013, pp. 101–106.

[19] G. Walsh, D. Tilbury, S. Sastry, R. Murray, and J.-P. Laumond, "Stabilization of trajectories for systems with nonholonomic constraints," *IEEE Transactions on Automatic Control*, vol. 39, no. 1, 1994.