This is the accepted version of a paper presented at 2017 American Control Conference, ACC 2017, Sheraton Seattle Hotel, Seattle, United States, 24 May 2017 through 26 May 2017.

Citation for the original published paper:

Distributed Cooperative Manipulation under Timed Temporal Specifications.
In: American Control Conference (ACC), 2017, 7963141 (pp. 1358-1363). IEEE
American Control Conference (ACC)

N.B. When citing this work, cite the original published paper.

Permanent link to this version:
http://urn.kb.se/resolve?urn=urn:nbn:se:kth:diva-215983
Distributed Cooperative Manipulation under Timed Temporal Specifications

Christos K. Verginis and Dimos V. Dimarogonas

Abstract—This paper addresses the problem of cooperative manipulation of a single object by \( N \) robotic agents under local goal specifications given as Metric Interval Temporal Logic (MITL) formulas. In particular, we propose a distributed model-free control protocol for the trajectory tracking of the cooperatively manipulated object without necessitating feedback of the contact forces/torques or inter-agent communication. This allows us to abstract the motion of the coupled object-agents system as a finite transition system and, by employing standard automata-based methodologies, we derive a hybrid control algorithm for the satisfaction of a given MITL formula. In addition, we use load sharing coefficients to represent potential differences in power capabilities among the agents. Finally, simulation studies verify the validity of the proposed scheme.

I. INTRODUCTION

Multi-agent systems have gained significant attention over the last decade, since they provide several advantages with respect to single-agent setups. In the case of object manipulation, complex tasks involving heavy/large payloads and difficult maneuvers necessitate the employment of more than one robot. The problem of cooperative manipulation control has been studied extensively, using centralized schemes, where a central computer handles the agents’ behavior, as well as decentralized setups, where each agent determines its actions on its own, either with partial or no communication at all [1]–[11].

In contrast to the related literature, which mainly considers the trajectory tracking of the manipulated object, we would like to define complex tasks over time, such as "never take the object to dangerous regions" or "keep moving the object from region A to B within a predefined time interval" which must be executed via the control actions of the agents. Such tasks can be expressed by temporal logic languages, which can describe complex planning objectives more efficiently than the well-studied navigation algorithms. Linear Temporal Logic (LTL) is the most common language that has been incorporated to the multi-agent motion planning problem [12]–[16], without however considering time specifications. Metric and Metric Interval Temporal Logic (MTL, MITL) [17]–[19] as well as Time Window Temporal Logic (TWTL) are languages that encode time specifications and were used for multi-agent motion planning in [20]–[22].

Regarding robotic manipulation, high level planning techniques have been proposed in [23]–[25] using common planning methods like configuration space potential fields and \( A^* \) algorithms. In [25] the motion planning problem for a group of unicycles manipulating a rigid body is addressed and in [24] an abstraction methodology is introduced; LTL specifications are employed in [26], where two mobile robots transport an object in a leader-follower scheme. Additionally, temporal logic formulas are utilized in [27] for dexterous manipulation through robotic fingers and in [28] for single manipulation tasks, without, however, incorporating the dynamics of the robotic arm in the abstracted model. In [29] a hybrid framework for cooperative manipulation is presented.

For the continuous control part, impedance and/or force control is the most common methodology [2]–[5], in which the robotic arms employ sensors to obtain feedback of the contact forces/torques which, however, may result to performance decline due to sensor noise or mounting difficulties. Moreover, most works in the related literature consider known dynamic models and/or parameters of the object and the agents, whose accurate knowledge, however, can be a challenging issue.

In this paper, we propose a novel hybrid control scheme for the cooperative manipulation of an object under MITL specifications. In particular, we develop a distributed model-free control protocol for the trajectory tracking of the cooperative manipulated object with prescribed transient and steady state performance. The latter allows us to abstract the motion of the coupled system object-agents as a finite transition system. Then, by employing formal verification-based methodologies, we derive a path that satisfies a given MITL task. The control scheme does not use any force/torque information at the contact points or any inter-agent communication and incorporates load sharing coefficients to account for differences in power capabilities among the agents.

The rest of the paper is organized as follows: Section II introduces notation and preliminary background. Section III describes the problem formulation and the overall system’s model. The control strategy is presented in Section IV. Section V verifies our approach through numerical simulation results and Section VI concludes the paper.

II. NOTATION AND PRELIMINARIES

A. Notation

The set of positive integers is denoted as \( \mathbb{N} \) and the real \( n \)-coordinate space, with \( n \in \mathbb{N} \), as \( \mathbb{R}^n \); \( \mathbb{R}_{\geq 0}^n \) and \( \mathbb{R}_{> 0}^n \) are the sets of real \( n \)-vectors with all elements nonnegative and positive, respectively. Given a set \( S \), \( 2^S \) is the set of all...
subsets of $S$. The vector connecting the origins of coordinate frames $\{A\}$ and $\{B\}$ expressed in frame $\{C\}$ coordinates in 3-D space is denoted as $p_{C/A}^B \in \mathbb{R}^3$. Given $a \in \mathbb{R}^3$, $S(a)$ is the skew-symmetric matrix defined according to $S(a)b = a \times b$. The rotation matrix from $\{A\}$ to $\{B\}$ is denoted as $R_{B/A} \in SO(3)$, where $SO(3)$ is the 3-D rotation group. The angular velocity of frame $\{B\}$ with respect to $\{A\}$ is denoted as $\omega_{B/A} \in \mathbb{R}^3$ and it holds that $[30] R_{B/A} = S(\omega_{B/A})R_{B/A}$. We further denote as $\eta_{A/B} \in \mathbb{R}^3$ the Euler angles representing the orientation of $\{B\}$ with respect to $\{A\}$ and $\omega_{B/A} = J_{A/B}(q_{A/B})\eta_{A/B}; J_{A/B} : T^3 \rightarrow \mathbb{R}^{3\times 3}$ is a smooth function representing the analytic Jacobian and $T^3$ is the 3-D torus. Moreover, $\mathcal{B}(c, r)$ denotes the 3-D sphere of radius $r \geq 0$ and center $c \in \mathbb{R}^3$ and $d : \mathbb{R}^3 \times \mathbb{R}^3 \rightarrow \mathbb{R}$ is the 3-D Euclidean distance. We further define the set $\mathcal{M} = \mathbb{R}^3 \times T^3$. For notational brevity, when a coordinate frame corresponds to an inertial frame of reference $\{I\}$, we will omit its explicit notation (e.g., $p_B = p_{B/I}^A, \omega_B = \omega_{B/I}^A, R_B = R_{B/I}$ etc.). Finally, all vector and matrix differentiations will be with respect to an inertial frame $\{I\}$, unless otherwise stated.

**B. Metric Temporal Logical Temporal Logic (MITL)**

The syntax of MITL over a set of atomic propositions $AP$ is defined by the grammar $\phi := p \mid \neg \phi \mid \bigcirc_i \phi \mid \Box_i \phi \mid \phi \cup \phi_2$, where $p \in AP, \bigcirc, \Box, \cup$ and $\cup$ are the next, future, always and until operators, respectively. $I$ is one of the following intervals: $[i_1, i_2), [i_1, i_2], (i_1, i_2), [i_1, \infty), (i_1, \infty)$ with $i_1, i_2 \in \mathbb{R}$ and $i_2 > i_1$. Given a set of atomic propositions $AP$, an MITL formula $\phi$ and an infinite sequence $r = (r_1, t_1)(r_2, t_2), \ldots$, with $r_j \in 2^{AP}$ and $t_j \in \mathbb{R}_0^\infty, t_{j+1} > t_j, \forall j \in \mathbb{N}$, we define $(r, j) = \phi, j \in \mathbb{N}$ (r satisfies $\phi$ at j) in the point-wise semantics [19]:

\[
(r, j) \models p \iff p \in r_j,
\]

\[
(r, j) \models \neg \phi \iff (r, j) \not\models \phi,
\]

\[
(r, j) \models \phi_1 \land \phi_2 \iff (r, j) \models \phi_1 \land (r, j) \models \phi_2,
\]

\[
(r, j) \models \bigcirc_i \phi \iff (r, j+1) \models \phi \land t_{j+1} - t_j \in I
\]

\[
(r, j) \models \phi \cup \phi_2 \iff \exists k, j, k \leq k, s.t. (r, k) \models \phi_2,
\]

\[
t_k - t_j \in I \land (r, m) \models \phi_1, \forall j \leq m \leq k.
\]

Also, $\bigcirc_i \phi = True \cup \phi$ and $\Box_i \phi = \neg \bigcirc_i \neg \phi$. The sequence $r$ satisfies $\phi$, denoted as $r \models \phi$ if and only if $(r, 1) \models \phi$. More information regarding MITL can be found in [18], [19].

**C. Dynamical Systems**

Consider the initial value problem:

\[
\dot{\xi} = H(t, \xi), \xi(t_0) = \xi^0 \in \Omega, \tag{1}
\]

with $H : [t_0, +\infty) \times \Omega \rightarrow \mathbb{R}^n$ where $\Omega \subset \mathbb{R}^n$ is a non-empty open set.

**Definition 1.** [31] A solution $\xi(t)$ of the initial value problem (1) is maximal if it has no proper right extension that is also a solution of (1).

**Theorem 1.** [31] Consider problem (1). Assume that $H(t, \xi)$ is: a) locally Lipschitz on $\xi$ for almost all $t \in [t_0, +\infty)$, b) piecewise continuous on $t$ for each fixed $\xi \in \Omega$, and c) locally integrable on $t$ for each fixed $\xi \in \Omega$. Then, there exists a maximal solution $\xi(t)$ of (1) on $[t_0, t_{\max}]$ with $t_{\max} > t_0$ such that $\xi(t) \in \Omega, \forall t \in [t_0, t_{\max}]$.

**Proposition 1.** [31] Assume that the hypotheses of Theorem 1 hold. For a maximal solution $\xi(t)$ on the time interval $[t_0, t_{\max}]$ with $t_{\max} < \infty$ and for any compact set $\Omega' \subset \Omega$, there exists a time instant $t' \in [t_0, t_{\max})$ such that $\xi(t') \notin \Omega'$.

**III. Problem Formulation**

Consider a bounded workspace $W \subset \mathbb{R}^3$ consisting of $N$ mobile manipulators rigidly grasping an object, as shown in Fig. 1. We assume that each agent $i \in \{1, \ldots, N\}$ has $n_i \geq 6$ degrees of freedom (generalized joint coordinates), denoted as $q_i : R_{\geq 0} \rightarrow \mathbb{R}^{n_i}$. We also denote the entire joint space as $\eta = [q_1^T, \ldots, q_N^T] : R_{\geq 0} \rightarrow \mathbb{R}^n$, with $n = \sum_{i=1}^{N} n_i$. The reference frames corresponding to the i-th end-effector and the object’s center of mass are denoted with $\{E_i\}$ and $\{O\}$, respectively, whereas $\{I\}$ corresponds to an inertial reference frame. The rigidity of the grasps implies that the agents can exert any forces/torques along every direction to the object. We consider that each agent $i$ knows the position and velocity only of its own joint variables $q_i$ as well as its own and the object’s geometric parameters. Moreover, no interaction force/torque measurements or online communication is required and the dynamic model of the object and the agents is considered unknown. Finally, we assume that the robotic agents and the object are away from kinematic and representation singularities [30].

**A. System model**

1) **Kinematics:** In view of Fig. 1, we have that

\[
p_{E_i}(\eta) = p_O(\eta) + p_{E_i/O}(\eta) = p_O(\eta) + R_{E_i/O}(q_i)
\]

\[
\eta_{E_i}(\eta) = \eta_O(\eta) + \alpha_i, \tag{2}
\]

where $\alpha_i \in T^3$ is a known angular offset. We further define $x_{E_i, x_O} : \mathbb{R}^n \rightarrow \mathbb{M}$ with $x_{E_i}(\eta) = [p_{E_i}^O(\eta), \eta_{E_i}^O(\eta)]^T$ and $x_{O}(\eta) = [p_{O}^O(\eta), \eta_{O}^O(\eta)]^T$. Note that, since (2) holds for all $i \in \{1, \ldots, N\}$, the physical coupling between the object and the agents creates a dependence of $x_O$ and $x_{E_i}$ on all $\eta$. Moreover, the grasp rigidity implies that $\omega_{E_i} = \omega_O$, i.e., $J_{E_i}(\eta_{E_i}(\eta))\eta_{E_i} = J_O(\eta_O(\eta))\eta_O$. Since the agents and the object are away from representation singularities, $J_{E_i}^{-1}$ and $J_{E_i}^{-1}$ are well-defined and smooth. Hence we differentiate (2) to obtain:

\[
\dot{x}_{E_i} = J_O(x_{E_i}, x_O)\dot{x}_O, \tag{3}
\]
where \( J_{O_i} : \mathbb{M} \times \mathbb{M} \to \mathbb{R}^{6 \times 6} \) is the Jacobian from the object to the \( i \)-th agent:

\[
J_{O_i} = \begin{bmatrix}
I_{3 \times 3} & S(p_{O_i/E})J_O(q_O) \\
0_{3 \times 3} & J_{Ei}^T(q_{Ei})J_O(q_O)
\end{bmatrix},
\]

which is always invertible due to the grasp rigidity.

Furthermore, by noticing that \( R_{E_i}(q)p_{Ei/o}^{E_i} = R_O(q)p_{Ei/o}^{E_i} \), (2) can be rewritten as

\[
x_{E_i} = f_{O_i}(x_o),
\]

where \( f_{O_i} : \mathbb{M} \to \mathbb{M} \) represents the coupled kinematics.

**Remark 1.** Each agent \( i \) can compute \( x_{E_i}, \dot{x}_{E_i} \) via its forward and differential kinematics [30]:

\[
x_{E_i} = k_i(q_i) \quad \text{and} \quad \dot{x}_{E_i} = J_i(q_i)\dot{q}_i,
\]

where \( k_i : \mathbb{R}^n \to \mathbb{R}^3 \) and \( J_i(q_i) = \partial k_i(q_i)/\partial q_i \) is the corresponding Jacobian. In addition, since the geometric parameters \( p_{Ei/o}^{E_i} \) and \( \alpha_i \) are known, \( x_O \) and \( \dot{x}_O \) can be computed by inverting eq. (2) and (3), without employing any sensory data.

### 2) Object Dynamics

The Newton-Euler equation for the object’s second-order dynamics is:

\[
M_O(x_O)\ddot{x}_O + C_O(x_O, \dot{x}_O)\dot{x}_O + g_O(x_O) + w_O(t) = \lambda_O.
\]

where \( M_O : \mathbb{M} \to \mathbb{R}^{6 \times 6} \) is the positive definite inertia matrix, \( C_O : \mathbb{M} \times \mathbb{R}^6 \to \mathbb{R}^{6 \times 6} \) is the Coriolis matrix, \( g_O : \mathbb{M} \to \mathbb{R}^6 \) is the gravity vector, \( w_O : \mathbb{R}\geq 0 \to \mathbb{R}^6 \) is a bounded vector representing external disturbances and \( \lambda_O \) is the force vector acting on the object’s center of mass. All aforementioned vector fields are continuous and unknown.

### 3) Agent Dynamics

The task-space dynamics for agent \( i \in \{1, \ldots, N\} \) are given by [30]:

\[
M_i(x_{E_i})\ddot{x}_{E_i} + C_i(x_{E_i}, \dot{x}_{E_i})\dot{x}_{E_i} + g_i(x_{E_i}) + f_i(x_{E_i}, \dot{x}_{E_i}) + w_i(t) + \lambda_i = u_i
\]

where \( M_i : \mathbb{M} \to \mathbb{R}^{6 \times 6} \) is the task-space positive definite inertia matrix, \( C_i : \mathbb{M} \times \mathbb{R}^6 \to \mathbb{R}^6 \) represents the task-space Coriolis matrix, \( g_i : \mathbb{M} \to \mathbb{R}^6 \) is the task-space gravity vector, \( f_i : \mathbb{M} \times \mathbb{R}^6 \to \mathbb{R}^6 \) is a vector field representing model uncertainties and \( w_i : \mathbb{R}\geq 0 \to \mathbb{R}^6 \) is a bounded vector representing external disturbances. Similarly to (6), the aforementioned vector fields are continuous and completely unknown; \( u_i \) is the task space wrench acting as the control input and \( \lambda_i \in \mathbb{R}^6 \) is the generalized force vector that agent \( i \) exerts on the object.

**Remark 2.** The task-space wrench \( u_i \) can be translated to the joint space inputs \( \tau_i \in \mathbb{R}^n \) via \( \tau_i = J_i^T(q_i)u_i + (I_{n \times n} - J_i^T(q_i)\dot{J}_i(q_i))\tau_o \), where \( \dot{J}_i \) is a generalized inverse of \( J_i \) [30]. The term \( \tau_o \) concerns redundant agents (\( n_i > 6 \)) and does not contribute to end-effector forces.

### 4) Coupled Dynamics

The kineto-statics duality [30] along with the grasp rigidity suggest that the force \( \lambda_O \) acting on the object center of mass and the generalized forces \( \lambda_i, i \in \{1, \ldots, N\} \) exerted by the agents at the contact points are related through

\[
\lambda_O = G^T(\pi)\lambda,
\]

where \( \lambda \) is the task-space wrench that agent \( i \) exerts on the object.

**Fig. 2.** (a): An example of the agents of Fig. 1 in the configuration that derives \( \hat{L} \). (b) The workspace partition according to the bounding box of the coupled system.
Next, we partition the workspace $W$ into $R$ equally sized rectangular regions $\Pi = \{\pi_1, \ldots, \pi_R\}$, whose geometric centers are denoted as $p^c_{\pi_j} \in W, j \in \{1, \ldots, R\}$. The length of the region sides is set as $D = 2\hat{L} + 2l_0$, where $l_0$ is an arbitrary positive constant. Hence, each region $\pi_j$ can be formally defined as follows:

$$\pi_j = \{p \in W \text{ s.t. } (p) \in [(p^c_{\pi_j})_k - \hat{L} - l_0, (p^c_{\pi_j})_k + \hat{L} + l_0], \forall k \in \{x, y, z\}\}.$$ 

with $d(p^c_{\pi_{j+1}}, p^c_{\pi_j}) = (2\hat{L} + 2l_0), \forall j \in \{1, \ldots, R - 1\}$ and $d(p^c_{\pi_1}, p^c_{\pi_R}) = (2\hat{L} + 2l_0)$. The notation $(a)_k, k \in \{x, y, z\}$ denotes the $k$-th coordinate of $a = [(a)_x, (a)_y, (a)_z]^T \in \mathbb{R}^3$. An illustration of the aforementioned partition is depicted in 2(b).

Note that each $\pi_j$ is a uniformly bounded, convex and well-connected set and also $\pi_j \cap \pi_{j'} = \emptyset, \forall j, j' \in \{1, \ldots, R\}$ with $j \neq j'$. We also define the neighborhood of region $\pi_j$ as the set of its adjacent regions $D(\pi_j) = \{\pi_{j'} \in \Pi \text{ s.t. } d(p^c_{\pi_j}, p^c_{\pi_{j'}}) = (2L + 2l_0)\}$, which is symmetric, i.e., $\pi_{j'} \in D(\pi_j) \iff \pi_j \in D(\pi_{j'})$.

To proceed we need the following definitions regarding the timed transition of the coupled system between two regions $\pi_j, \pi_{j'}$:

**Definition 2.** The coupled system object-agents is in region $\pi_j$ at a configuration $q$, denoted as $A(q) \in \pi_j$, if and only if (i) $p_s \in \pi_j, \forall p_s \in S_{\pi_j}$ and (ii) $d(p_0(q), p^c_{\pi_j}) < l_0$.

**Definition 3.** Assume that $A(q(t_0)) \in \pi_j, j \in \{1, \ldots, R\}$, for some $t_0 \in \mathbb{R}_{\geq 0}$. Then, there exists a transition for the coupled system object-agents from $\pi_j$ to $\pi_{j'} \in \{1, \ldots, R\}$ with time duration $\delta t_{j,j'} \in \mathbb{R}_{\geq 0}$, denoted as $\pi_j \rightarrow \pi_{j'}$, if and only if there exists a bounded control trajectory $\pi$ in (10), such that $A(q(t_0 + \delta t_{j,j'})) \in \pi_{j'}'$ and $p_s \in \pi_j \cup \pi_{j'}, \forall p_s \in S_{\pi_j}, t \in [t_0, t_0 + \delta t_{j,j'}]$.

Note that the entire system object-agents must remain in $\pi_j, \pi_{j'}$ during the transition and therefore the requirement $\pi_{j'} \in D(\pi_j)$ is implicit in the definition above.

**C. Specification**

Given the workspace partition, we can introduce a set of atomic propositions $\mathcal{AP}$ for the object, which are expressed as boolean variables that correspond to properties of interest in the regions of the workspace (e.g., "Obstacle region", "Goal region"). Formally, the labeling function $\mathcal{L} : \Pi \rightarrow 2^{\mathcal{AP}}$ assigns to each region $\pi_j$ the subset of the atomic propositions $\mathcal{AP}$ that are true in $\pi_j$.

**Definition 4.** Assume that $q(t)$ is a trajectory. Then, a timed sequence of $q$ is the infinite sequence $\beta = (q_1(t), t_1) \gamma_2(t_2), \ldots$ with $t_m \in \mathbb{R}_{\geq 0}, t_{m+1} > t_m$ and $A(q_m(t_m)) \in \pi_{j_m}, j_m \in \{1, \ldots, R\}, \forall m \in \mathbb{N}$. The timed behavior of $\beta$ is the infinite sequence $\sigma_\beta = (\sigma_1, t_1) \sigma_2(t_2), \ldots$ with $\sigma_m \in 2^{\mathcal{AP}}, \sigma_m \in \mathcal{L}(\pi_{j_m})$ for $A(q_m(t_m)) \in \pi_{j_m}, j_m \in \{1, \ldots, R\}, \forall m \in \mathbb{N}, i.e., the set of atomic propositions that are true when $A(q_m(t_m)) \in \pi_{j_m}$.

**Definition 5.** The timed run $\beta$ satisfies an MITL formula $\phi$ if and only if $\sigma_\beta \models \phi$.

![Fig. 3](image-url) A transition between two adjacent regions $\pi_j$ and $\pi_{j'}$. Since $p_0 \in B(p_0(t), l_0)$, we conclude that $p_s \in B(p_0, \hat{L}) \in B(p_d(t), l_0 + L) \in \pi_j \cup \pi_{j'}$.

We are now ready to state the problem treated in this paper.

**Problem 1.** Given $N$ agents rigidly grasping an object in $W$ subject to the coupled dynamics described by (10), the workspace partition $\Pi$ such that $A(q(0)) \in \pi_j, j \in \{1, \ldots, R\}$, a MITL formula $\phi$ over $\mathcal{AP}$ and the labeling function $\mathcal{L}$, develop a control strategy that achieves a timed sequence $\beta$ which yields the satisfaction of $\phi$.

**IV. MAIN RESULTS**

**A. Control Design**

The first ingredient of the proposed solution, which constitutes one of the main novelties of this work, is the design of a decentralized control protocol $\tau$ such that a transition relation between two adjacent regions according to Def. 3 is established. Assume, therefore, that $A(q(t_0)) \in \pi_j(t_0) \in \mathbb{R}_{\geq 0}$. We wish to find a bounded $\tau$, such that (i) $A(q(t_0 + \delta t_{j,j'})) \in \pi_{j'}$, with $\pi_{j'} \in D(\pi_j)$ and (ii) $p_s \in \pi_j \cup \pi_{j'}, \forall p_s \in S_{\pi_j}, t \in [t_0, t_0 + \delta t_{j,j'}]$, for a predefined arbitrary constant $\delta t_{j,j'} \in \mathbb{R}_{\geq 0}$ corresponding to the transition $\pi_j \rightarrow \pi_{j'}$.

The first step is to associate to the transition a smooth trajectory defined by the line segment that connects $p^c_{\pi_j}$ and $p^c_{\pi_{j'}}$, i.e. define $p_d : [t_0, \infty) \rightarrow \mathbb{R}^3$, such that $p_d(t_0) = p^c_{\pi_j}, p_d(t) = p^c_{\pi_{j'}}, \forall t \geq t_0 + \delta t_{j,j'}$ and

$$\mathcal{B}(p_d(t), \hat{L} + l_0) \in \pi_j \cup \pi_{j'}, \forall t \geq t_0. \tag{12}$$

If we guarantee that the object’s center of mass stays close to $p_d$, i.e., $d(p_0 + \delta t_{j,j'}, p^c_{\pi_{j'}}) \in \mathbb{R}_{\geq 0}$ and by invoking (11) and (12), we obtain $p_s \in B(p_0(t_0), \hat{L}) \in B(p_d(t), \hat{L} + l_0) \in \pi_j \cup \pi_{j'}, \forall p_s \in S_{\pi_j}, t \geq t_0$ (and therefore $t \in [t_0, t_0 + \delta t_{j,j'}]$, and thus the requirements of Def. 3 for the transition relation are met. Fig. 3 illustrates the aforementioned reasoning. Along with $p_d$, we also define a desired smooth trajectory $\eta_d : [t_0, \infty) \rightarrow \mathbb{R}^3$ for the object orientation and we form the desired pose trajectory $x_d : [t_0, \infty) \rightarrow \mathbb{M}$, with $x_d(t) = [x_{d_3}(t), \ldots, x_{d_R}(t)]^T = [p_{d_3}^c(t), \eta_d(t)]^T$. We can now define the associated pose errors $e_s : [t_0, \infty) \rightarrow \mathbb{M}$ with

$$e_s(t) = [e_{s_1}(t), \ldots, e_{s_R}(t)]^T = x_o(q(t)) - x_d(t). \tag{13}$$

A suitable methodology for the control design in hand is that of prescribed performance control, recently proposed in [32], which is adapted in this work in order to achieve predefined transient and steady state response bounds for the pose errors.
As stated in [32], prescribed performance characterizes the behavior where the aforementioned errors evolve strictly within a predefined region that is bounded by absolutely decaying functions of time, called performance functions. The mathematical expressions of prescribed performance are given by the inequalities: $-\rho_{st}(t) < \varepsilon_{st}(t) < \rho_{st}(t), \forall k \in \{1, \ldots, 6\}$, where $\rho_{st} : [t_0, \infty) \rightarrow \mathbb{R}^6_+ \supset \rho_{st}(t) = (\rho_{st}^1 - \rho_{st}^e \varepsilon(t - t_0) + \rho_{st}^0)$ and $\varepsilon_{st}(t)$ and $\rho_{st}(t)$ are designer-specified, smooth, bounded and decreasing positive functions of time with positive parameters $l_{st}, \rho_{st}^e$, incorporating the desired transient and steady state performance respectively. In particular, the decreasing rate of $\rho_{st}(t)$, which is affected by the constant $l_{st}$, introduces a lower bound on the speed of convergence of $\varepsilon_{st}(t)$. Furthermore, the constants $\rho_{st}^e$ can be set arbitrarily small, achieving thus practical convergence of the pose errors to zero.

Next, we propose a state feedback control protocol that does not incorporate any information on the agents’ or the object’s dynamics or the external disturbances and guarantees that $\varepsilon_{st}(t) < \rho_{st}(t)$ for all $[t_0, \infty)$ and hence $[t_0, t_0 + \delta t_{j'}], \forall k \in \{1, 2, 3\}$, $\rho_{st}^e > |\varepsilon_{st}(t)|, \forall k \in \{4, 5, 6\}$ and $\rho_{st}^e > \rho_{st}^0 > 0, l_{st} > 0, \forall k \in \{1, \ldots, 6\}$.

**Step I-a.** Select the corresponding functions $\rho_{st}(t) = (\rho_{st}^1 - \rho_{st}^e \varepsilon(t - t_0) + \rho_{st}^0), \forall k \in \{1, 2, 3\}$, $\rho_{st}^e > |\varepsilon_{st}(t)|, \forall k \in \{4, 5, 6\}$ and $\rho_{st}^e > \rho_{st}^0 > 0, l_{st} > 0, \forall k \in \{1, \ldots, 6\}$.

**Step I-b.** Define the normalized errors $\xi_{st} : [t_0, \infty) \rightarrow \mathbb{R}$:

$$\xi_{st}(t) = \rho^{-1}_{st}(t)\varepsilon_{st}(t), \forall k \in \{1, \ldots, 6\}, \quad (14)$$

and design the reference velocity vector as $\dot{x}^* : \mathbb{R}^6 \times [t_0, \infty) \rightarrow \mathbb{R}^6$, with $\dot{x}^* = [\dot{x}^*_{d1}, \ldots, \dot{x}^*_{d6}]^T$ and:

$$\dot{x}^*_d(\xi_{st}, t) = -g_{st}\xi_{st}(\xi_{st}), \forall k \in \{1, \ldots, 6\}, \quad (15)$$

where $g_{st} > 0$ and $\xi_{st} : \mathbb{R} \rightarrow \mathbb{R}$ is the transformed error:

$$\xi_{st}(\xi_{st}) = \ln(1 + \frac{1}{\xi_{st}}), \forall k \in \{1, \ldots, 6\}. \quad (16)$$

Before proceeding, note that (13) and (14) suggest that $x_o = P(t)\xi_o$, where $\xi_o = [\xi_{o1}, \ldots, \xi_{o6}]^T$ and $P_o = \text{diag}([\rho_{st}|_{k \in \{1, \ldots, 6\}}])$. Therefore, by employing (5) we can write $\mathcal{X} = [\mathcal{X}^*, f_o^*(x_0), \ldots, f_o^*(x_0)]^T = [\xi^* P_o(t) + x_d^T(t), f_o^*(P_o(t)\xi_o + x_d(t)), \ldots, f_o^*(P_o(t)\xi_o + x_d(t))]^T = \mathcal{T}_o(P_o(t)\xi_o + x_d(t))$, i.e., we can express $\mathcal{X}$ as a function of $\xi_o$ and $t$. Therefore, in the following, the dependence on $\mathcal{X}$ will directly imply dependence on $\xi_o, t$. Similarly, dependence on $\mathcal{X}$ will imply dependence on $\xi_o, t$ and $\mathcal{X}$.

**Step II-a.** Define the velocity error vector $\xi_o : \mathbb{R}^6 \times [t_0, \infty) \rightarrow \mathbb{R}^6$ with $\xi_o(\xi_o, t)$:

$$\xi_o(t) = [\xi_{o1}(t), \ldots, \xi_{o6}(t)]^T = \dot{x}_o(\mathcal{X}(t)) - \dot{x}_o^*(\xi_o, t)$$

and select the corresponding positive performance functions $\rho_{o\xi} : [t_0, \infty) \rightarrow \mathbb{R}^6_+$ with $\rho_{o\xi}(t) = (\rho_{o\xi}^1 - \rho_{o\xi}^e \varepsilon(t_0) + \rho_{o\xi}^0)$, such that $\rho_{o\xi}^1 > \varepsilon_{o\xi}(t_0), \varepsilon_{o\xi} > 0$ and $\rho_{o\xi}^e > \rho_{o\xi}^0 > 0, \forall k \in \{1, \ldots, 6\}$.

**Step II-b.** Define the normalized velocity errors $\xi_v : [t_0, \infty) \rightarrow \mathbb{R}^6$:

$$\xi_v(t) = [\xi_{v1}(t), \ldots, \xi_{v6}(t)]^T = P_o^{-1}(t)e_o(\mathcal{X}(t), t), \quad (17)$$

where $P_v(t) = \text{diag}([\rho_{sv}(t)|_{k \in \{1, \ldots, 6\}}])$, and design the distributed control protocol $u_i : \mathbb{R}^6 \times \mathbb{R}^6 \times [t_0, \infty) \rightarrow \mathbb{R}^6$:

$$u_i(\xi_i, \xi_v, t) = -c_i g_{J_{oi}}^T(\mathcal{X}) P_v^{-1}(t) R_v(\xi_v) e_v(\xi_v), \quad (18)$$

where $g_{J_{oi}} > 0, J_{oi}$ as defined in (4) and $c_i$ are predefined load sharing coefficients satisfying $\sum_{i \in \{1, \ldots, N\}} c_i = 1$ and $0 < c_i \leq 1, \forall i \in \{1, \ldots, N\}$. Also, $R_v : \mathbb{R}^6 \rightarrow \mathbb{R}^6 \times \mathbb{R}^6 \times \varepsilon_v : \mathbb{R}^6 \rightarrow \mathbb{R}^6 \times \mathbb{R}^6$ are defined as:

$$R_v(\varepsilon_v) = \text{diag}\left\{\begin{array}{c} (2(1 - \xi_v^2)^{-1}) \sum_{k \in \{1, \ldots, 6\}} \\ \frac{1 + \xi_v}{1 - \xi_v} \end{array}\right\} \quad (19)$$

$$\varepsilon_v(\varepsilon_v) = \left[\frac{1 + \xi_v}{1 - \xi_v}, \ldots, \frac{1 + \xi_v}{1 - \xi_v}\right]^T \quad (20)$$

The control law (18) can be written in vector form $\mathbf{v}(\mathbf{u}_1, \ldots, \mathbf{u}_N)^T$:

$$\mathbf{v}(\xi_i, \xi_v, t) = U_j^T(t) = -C_i G^*(\mathcal{X}) P_v^{-1}(t) R_v(\xi_v) e_v(\xi_v), \quad (21)$$

where $C_i = g_{J_{oi}} \text{diag}([c_i, l_{o\xi}, \xi_{o\xi}]) \in \mathbb{R}^{6N \times 6N}$, $G^*(\mathcal{X}) = [J_{o1}^{-1}, \ldots, J_{oN}^{-1}]^T \in \mathbb{R}^{6N \times 6N}$ and the notation $U_j^T$ stands for the transition from $\mathbf{v}_j$ to $\mathbf{v}_j$.

**Remark 3.** It can be verified by (15) and (18) that the proposed control protocol is distributed in the sense that each agent needs feedback only from the state of the object’s center of mass, which can be obtained by (2) and (3), as discussed in Section III-A. The parameters needed for the computation of $\rho_{st}, \rho_{o\xi}, \eta_d$ as well as $g_{J_{oi}}, c_i, k \in \{1, \ldots, 6\}, i \in \{1, \ldots, N\}$ can be transmitted off-line to the agents. Moreover, the overall control scheme does not incorporate any prior knowledge of the model nonlinearities/disturbances or force/torque measurements at the contact points.

**Remark 4.** Note that internal force regulation can be also guaranteed by including in (21) a term of the form $(I_{6N} - \frac{1}{c_i} G^T G^T) f_{i\mathbf{d}_j}$, which reduces to knowledge of the offsets $p_{ei}/\xi$ (since the agents can compute $R_v$ and $P_v$), that can be also transmitted off-line to the agents.

The next theorem summarizes the results of this section.

**Theorem 2.** Consider $N$ agents rigidly grasping an object with unknown coupling dynamics described by (10) and (11) and $\mathcal{A}(\mathcal{Q}(t_0)) \in \pi_j, j \in \{1, \ldots, R\}$. Then, the distributed control protocol (14)-(20) guarantees that $\pi_j \rightarrow \pi_j^*$ with time duration $\delta_{j,j'}$, and all closed loop signals being bounded, and thus establishes a transition relation between $\pi_j, \pi_j^*$ for the coupled system object-agents, according to Def. 3.

Differentiating (14) and (17) with respect to time, employing (13), (10) as well as the facts that $\dot{x}_o = \dot{x}_o^* + \dot{\mathcal{X}}$.
\[ P_v(t)\varepsilon_k + \sum_{j=1}^{N-1} c_j = 1 \] and substituting (15), (21), we obtain:
\[
\dot{\xi}_K(t) = h_{\xi}(t) + \delta_{\xi}(t) + \rho_{\xi}(t)\varepsilon_k(t) - \rho_{\xi}(t)\varepsilon_k(t), \forall k \in \{1, \ldots, 6\} \tag{22}
\]
\[
\dot{\xi}_n(t) = h_{\xi}(t) + \delta_{\xi}(t) + \rho_{\xi}(t)\varepsilon_k(t) - \rho_{\xi}(t)\varepsilon_k(t), \forall k \in \{1, \ldots, 6\} \tag{22}
\]
for all \( t \geq t_0 \) with \( \varepsilon_k(t) = [\varepsilon^T_k(t, \xi(t)], \varepsilon^T_k(t, \xi(t))] \). By defining \( h : \mathbb{R}^6 \times \mathbb{R}^6 \times [0, \infty) \rightarrow \mathbb{R}^{12} \), we can write (22), (23) in compact form:
\[
\dot{\xi} = h(\xi(t), \xi(t), \xi(t)) \tag{24}
\]
for all \( t \geq t_0 \) with \( \varepsilon_k(t) = [\varepsilon^T_k(t, \xi(t)], \varepsilon^T_k(t, \xi(t))] \).

We further define the open and nonempty set \( \Omega_\xi = \Omega_{\xi_1} \times \Omega_{\xi_2} \subset \mathbb{R}^{12} \) with \( \Omega_{\xi_1} = \Omega_{\xi_2} = (-1, 1)^9 \).

Since \( \Delta \overline{\sigma}(\xi_k, t) \in \pi_j \), Def. 2 implies that \( d(p_0(\xi_k, t), p_{\xi_j}) \leq 1 \).

Also, \( p_0(\xi_k, t) = p_{\xi_j} \).

Therefore, by choosing \( \rho_{\xi_k} = \lambda_0, \forall k \in \{1, 2, 3\} \), we guarantee that \( \bar{\xi}_k(t) \in \Omega_{\xi_k} \). Furthermore, by selecting \( \rho_{\xi_j} = |\varepsilon_{\xi_j}(t)|, \forall k \in \{1, 2, 3\} \), we guarantee that \( \bar{\xi}_n(t) \in \Omega_{\xi_n} \). Thus, we conclude that \( \bar{\xi}_n(t) \in \Omega_{\xi_n} \).

Additionally, \( h \) is continuous on \( t \) and locally Lipschitz on \( \xi \) over \( \Omega_\xi \). Therefore, according to Theorem 1, there exists a maximal solution \( \xi(t) \) of (24) on a time interval \([t_0, t_{\text{max}}]\) such that \( \xi(t) \in \Omega_\xi, \forall t \in [t_0, t_{\text{max}}] \).

Thus:
\[
\xi(t) = \frac{\varepsilon_{\text{m}}}{\rho_{\text{m}}} \in (-1, 1), \forall k \in \{1, 2, 3\}, \forall \varepsilon_k \in \{1, 2, 3\}.
\]

From (20), we obtain:
\[
\varepsilon_{\text{m}}(t) = \frac{\varepsilon_{\text{m}}}{\rho_{\text{m}}} \in (-1, 1), \forall k \in \{1, 2, 3\}, \forall \varepsilon_k \in \{1, 2, 3\}.
\]

Next, since \( \dot{x}_k \), \( \rho_{\xi_k} \), \( \dot{\rho}_{\xi_k} \) are bounded by construction and \( \xi_k \), \( \xi_k \) are bounded in \( (-1, 1) \), we can change (25) to obtain:
\[
V_{\text{v}} = \frac{4\varepsilon_{\text{m}}}{1 - \xi_k^2}(g_{\xi_k} \varepsilon_k + \dot{x}_k(\xi_k) - \rho_{\xi_k}(\xi_k) - \rho_{\xi_k}(\xi_k)\varepsilon_k)
\]

Next, since \( \dot{x}_k \), \( \rho_{\xi_k} \), \( \dot{\rho}_{\xi_k} \) are bounded by construction and \( \xi_k \), \( \xi_k \) are bounded in \( (-1, 1) \), we change (25) to obtain:
\[
V_{\text{v}} = \frac{4\varepsilon_{\text{m}}}{1 - \xi_k^2}(g_{\xi_k} \varepsilon_k + \dot{x}_k(\xi_k) - \rho_{\xi_k}(\xi_k) - \rho_{\xi_k}(\xi_k)\varepsilon_k)
\]

From (20), we obtain:
\[
|\varepsilon_k|_k(t) = \max\{|\varepsilon_k|_k(t)|, g_{\xi_k} \varepsilon_k - \dot{x}_k(\xi_k)|, \frac{\rho_{\xi_k}(\xi_k)}{1 - \xi_k^2}
\]

From (20), we obtain:
\[
|\varepsilon_k|_k(t) = \max\{|\varepsilon_k|_k(t)|, g_{\xi_k} \varepsilon_k - \dot{x}_k(\xi_k)|, \frac{\rho_{\xi_k}(\xi_k)}{1 - \xi_k^2}
\]

From (20), we obtain:
\[
|\varepsilon_k|_k(t) = \max\{|\varepsilon_k|_k(t)|, g_{\xi_k} \varepsilon_k - \dot{x}_k(\xi_k)|, \frac{\rho_{\xi_k}(\xi_k)}{1 - \xi_k^2}
\]

From (20), we obtain:
\[
|\varepsilon_k|_k(t) = \max\{|\varepsilon_k|_k(t)|, g_{\xi_k} \varepsilon_k - \dot{x}_k(\xi_k)|, \frac{\rho_{\xi_k}(\xi_k)}{1 - \xi_k^2}
\]
Remark 5. Instead of employing the control protocol (21) over \( \{t_0, \infty\} \), we can define it over a finite time interval as \( \pi = U_{\pi_j}^j \left( \{t_0, t_0 + \delta \tau_j\} \right) \). In that case, it follows by the continuity of \( d, p_0, p_d \) that \( \lim_{t \to \infty} d(t) = d(p_0(t), p_d(t)) = d(p_0(t_0 + \delta \tau_j), p_d(t_0 + \delta \tau_j)) \) and therefore, the transition \( \pi_j \to \pi_j' \) with time duration \( \delta \tau_j \) is still achieved. Moreover, the predefined selection of \( \delta \tau_j \) for each transition \( \pi_j \to \pi_j' \) is related to the control capabilities of the agents, since smaller \( \delta \tau_j \) will produce larger, but still bounded, \( \dot{x}_d \) and \( \pi \).

B. High-Level Plan Generation

The second part of our solution is the derivation of a high-level plan that satisfies the given MITL formula \( \phi \) and can be generated using standard techniques from automata-based formal verification methodologies. Thanks to our proposed control law that allows the transition \( \pi_j \to \pi_j' \) for all \( \pi_j \in \Pi \) with \( \pi_j' \in D(\pi_j) \) in a predefined time interval \( \delta \tau_j \), we can abstract the motion of the coupled system object-agents as a finite weighted transition system (WTS) \([33]\) \( \mathcal{T} = \{\Pi, \Pi_0, \to, \mathcal{AP}, \mathcal{L}, \gamma\} \), where \( \Pi \) is the set of states defined in Section III-B, \( \Pi_0 \subset \Pi \) is a set of initial states, \( \to \subset \Pi \times \Pi \) is a transition relation according to Def. 3, \( \mathcal{AP} \) and \( \mathcal{L} \) are the atomic propositions and the labeling function, respectively, as defined in Section III-C, and \( \gamma : \to \to \mathbb{R} > 0 \) is a map that assigns to each transition its time duration, i.e., \( \gamma(\pi_j \to \pi_j') = \delta \tau_j \). Therefore, by designing the switching control protocol \( U_{\tau_j}^{j+1} \) (from (21)):

\[
\pi = U_{\tau_j}^{j+1}(t), \forall t \in [t_j, t_j + \delta \tau_{j,r_{j,i}},) \), \( j \in \mathbb{N}, \number{29}
\]

with (i) \( t_j = 0 \), (ii) \( t_{j+1} = t_j + \delta \tau_{j,r_{j,i}} \) and (iii) \( r_j \in \{1, \ldots, R\}, \forall j \in \mathbb{N} \), we can define a timed run \( \pi \) of the WTS as an infinite sequence \( r = (\pi_{r_1}, t_1)(\pi_{r_2}, t_2), \ldots \), where \( \pi_{r_i} \in \Pi_0 \) with \( \mathcal{A}(\pi(0)) \in \pi_{r_1}, \pi_{r_j} \in \Pi, r_j \in \{1, \ldots, R\} \) and \( t_j \) are the corresponding time stamps such that \( \mathcal{A}(\pi(t_j)) \in \pi_{r_j}, \forall j \in \mathbb{N} \). Every timed run \( r \) generates a timed word \( w(r) = (\mathcal{L}(\pi_{r_1}), t_1)(\mathcal{L}(\pi_{r_2}), t_2), \ldots \) over \( \mathcal{AP} \) where \( \mathcal{L}(\pi_{r_j}), j \in \mathbb{N} \) is the subset of the atomic propositions \( \mathcal{AP} \) that are true when \( \mathcal{A}(\pi(t_j)) \in \pi_{r_j} \).

The given MITL formula \( \phi \) is translated into a Timed Büchi Automaton \( \mathcal{A}_{\phi} \) [17] and the product \( \mathcal{A}_p = \mathcal{T} \otimes \mathcal{A}_{\phi} \) is built [33]. The projection of the accepting runs of \( \mathcal{A}_p \) onto \( \mathcal{T} \) provides a timed run \( r_\phi \) of \( \mathcal{T} \) that satisfies \( \phi \); \( r_\phi \) has the form \( r_\phi = (\pi_{r_1}, t_1)(\pi_{r_2}, t_2), \ldots \), i.e., an infinite \(^1\) sequence of regions \( \pi_{r_i} \), to be visited at specific time instants \( t_j \) (i.e., \( \mathcal{A}(\pi(t_j)) \in \pi_{r_j}, \forall j \in \mathbb{N} \)). More details on the technique are beyond the scope of this paper and the reader is referred to [17, 20, 33].

The execution of \( r_\phi = (\pi_{r_1}, t_1)(\pi_{r_2}, t_2), \ldots \) produces a trajectory \( \pi(t), t \in \mathbb{R} > 0 \), with timed sequence \( \beta_\phi = (\pi_{r_1}(t), t_1)(\pi_{r_2}(t), t_2), \ldots \) with \( \mathcal{A}(\pi(t_j)) \in \pi_{r_j}, \forall j \in \mathbb{N} \). Following Def. 4, \( \beta_\phi \) has the timed behavior \( \sigma_{\beta_\phi} = (\pi_{r_1}, t_1)(\pi_{r_2}, t_2), \ldots \) with \( \sigma_j \in \mathcal{L}(\pi_{r_j}) \), for \( \mathcal{A}(\pi(t_j)) \in \mathcal{AP} \).

\(^1\)It can be proven that if such a run exists, then there also exists a run that can be always represented as a finite prefix followed by infinite repetitions of a finite suffix [33].
the cooperative manipulation of an object by $N$ agents considering non-rigid grasps and compensating for uncertain geometric characteristics of the objects.

Regarding each transition $\pi_{r_j} \rightarrow \pi_{r_{j+1}}, j \in \{1, \ldots, 10\}$, we chose $\eta_d = [0, 0, 0]^T$, $\rho_{sk} = l_0 = 0.5$, $\rho_{sk}^\infty = 0.01$ and $l_{sk} = 1$ as well as $\rho_{uk} = 2|c_{uk}(t_j)| + 0.1$, $\rho_{uk}^\infty = 0.01$ and $l_{uk} = 1, \forall k \in \{1, \ldots, 6\}$. The control gains were chosen as $g_{sk} = 0.1$ and $g_{v} = 2.5, \forall k \in \{1, \ldots, 6\}$ to retain the required input signals $u_i$ within feasible ranges that can be implemented by real actuators. Finally, $w_o, w_i$ and $f_i$ were chosen as sinusoidal functions of time and the load sharing coefficients were selected as $c_1 = 0.5, c_2 = 0.35$ and $c_3 = 0.15$ to demonstrate a potential difference in the power capabilities of the agents. Fig. 5 depicts the transitions of the coupled system object-agents. It can be deduced from the figure that $p_o \in \mathcal{B}(p_o(t), l_0), \forall t \in [0, 50]$ and therefore $p_s \in \pi_{r_j} \cup \pi_{r_{j+1}}, \forall p_s \in \mathcal{S}, j \in \{1, \ldots, 10\}$, verifying thus the theoretical results. Moreover, Fig. 6 shows the control inputs of the three agents, demonstrating the effect of the load sharing coefficients.

VI. CONCLUSION AND FUTURE WORK

In this work we proposed a hybrid control strategy for the cooperative manipulation of an object by $N$ agents under MITL specifications. In particular, we developed a robust decentralized control protocol that allowed us to abstract the motion of the coupled system object-agents as a finite transition system. Then, we employed standard formal-verification tools for the derivation of a path that satisfied the high level goal. Future efforts will be devoted towards considering non-rigid grasps and compensating for uncertain geometric characteristics of the objects.

REFERENCES


planning of multi-agent systems under timed temporal specifications,”

with applications to multi-uav mission planning,” Intern. Journal of
Robust and Nonlinear Control, 2011.

[22] D. Aksaray, C.-I. Vasile, and C. Belta, “Dynamic routing of energy-

multiple mobile robots for cooperative manipulation and transporta-

cooperative multiple robot planar manipulation,” Robotics: Science

a group of micro agents manipulating a rigid object,” in MED, 2005.

Dimarogonas, and K. J. Kyriakopoulos, “Decentralized leader-follower
control under high level goals without explicit communication,” in

Murray, “Towards formal synthesis of reactive controllers for dexter-
ous robotic manipulation,” in ICRA, 2012.

manipulation planning with temporal logic specifications,” in Intern.
Conf. on Robotics and Automation, 2015.

[29] L. Chaimowicz, M. Campos, and V. Kumar, “Hybrid systems modeling


[31] E. D. Sontag, Mathematical control theory: deterministic finite di-

approximation-free control scheme with prescribed performance for
unknown pure feedback systems,” Automatica, 2014.