Motion Tasks for Robot Manipulators on Embedded 2-D Manifolds under Input Constraints

Xanthi Papageorgiou, Herbert G. Tanner and Kostas J. Kyriakopoulos

***Abstract*— We present a methodology to steer the end effector of a robotic manipulator, which is constrained in terms of joint rates, on the surface within the workspace. We develop controllers for stabilizing the end effector to a point, and for tracking a trajectory on this surface, while respecting the input constraints. We show that the resulting closed loop system is uniformly asymptotically stable and we verify our analytical development with computer simulations.**

I. Introduction

Robotic applications where the manipulator is supposed to perform a task along a particular surface, such as robotic surface painting, surface cleaning, and surface inspection, pose challenging control design problems. Our motivation comes from the field of neuro-robotics, and specifically from an application where a robot executes a task through interfacing with the neural system (Fig. 1), thus by processing electromyographic activity. In most cases, neural signals are noisy and inappropriate for controlling a robot directly. The presense of obstacles in the environment, and consideration of non-planar surfaces complicates the problem further. We need a strategy to combine compliant behavior of the robot with respect to its environment, and obstacle avoidance.

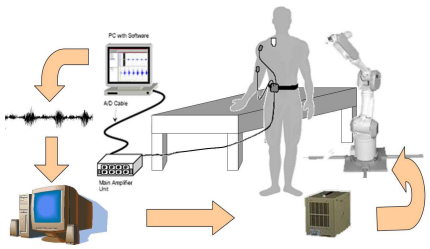

Fig. 1. The problem motivation.

Related previous relevant work has focused on the problem of automotive painting of surfaces that are convex and have no holes, [1], [2], [3]. In [1], the authors decompose the coverage trajectory generation problem into three subproblems: selection of the start curve, selection of the speed profiles along each pass, and selection of the spacing between the passes.

This work is partially supported by the European Commission through contract "FP6 - IST - 001917 - NEUROBOTICS: The fusion of Neuroscience and Robotics", and by Eugenides Foundation Scholarship.

X. Papageorgiou and K.J. Kyriakopoulos are with the School of Mechanical Engineering, National Technical University of Athens, Athens, Greece, xpapag@mail.ntua.gr; kkyria@mail.ntua.gr

H.G. Tanner is with the Department of Mechanical Engineering, MSC01 1150, the University of New Mexico, Albuquerque, NM 87131, USA, tanner@unm.edu

In previous work [4], we presented a methodology to drive the end-effector of a non-redundant manipulator to a surface while avoiding obstacles. Once the end-effector is in close proximity of the surface, a second controller takes over to stabilize the end-effector at a predefined distance to the surface. Motion planning and tracking tasks are then considered, without however taking into account kinematic input constraints.

Input constraints in the context of nonlinear systems have been recently treated in the framework of multi-robot motion planning using navigation functions [5], [6]. In [7] the input constraints are being enforced by design in the form of a hybrid system. In [8], another switching controller is introduced that implements specific favorable velocity profiles on multiple micro-robots.

In this paper, we consider the control design problem for a kinematically redundant manipulator, the joint rate inputs of which must remain within pre-specified bounds. We do so by building navigation functions [5], [6] and analyzing the closed loop system that has saturated inputs by means of nonsmooth stability analysis. The system switches between different controllers when it finds itself within certain regions of the workspace (called belt zones [9], [10]). The contribution of this paper is the development of globally uniformly asymptotically stable controllers for redundant articulated robot manipulators, subject to input constraints, to achieve

- reference trajectory tracking with obstacle avoidance on 2-D manifolds embedded in 3-D workspaces, and
- stabilization with obstacle avoidance on 2-D manifolds embedded in 3-D workspaces.

II. Problem Statement

Considering the motion planning problem of a redundant robotic manipulator, with kinematic input constraints, in a workspace with obstacles. The objective is for the robot to move near the surface, and track a predefined trajectory on it. We assume that we have a stationary environment and that we have direct control on the manipulator joint rates. Thus the robot can be kinematically described by a set of integrators:

$$\dot{q} = u \tag{1}$$

where $q = [q_1 \ \dots \ q_m]^T \in \mathbb{R}^m$ is the vector of arm joint variables, and u the joint velocity inputs. Let the admissible and feasible configuration space (workspace) for the manipulator be denoted $\mathcal{W} \subset \mathbb{R}^m$. The obstacle free subset of the workspace is denoted $\mathcal{W}_{free} \subseteq \mathcal{W}$. Let $\mathcal{O} \in \mathcal{W} \setminus \mathcal{W}_{free}$ be

Proc. European Control Conf. (ECC-2007), Kos, Greece, July 2-5, 2007, p.p. 3783-3789

the set of all obstacles in 3-D workspace. Define a vector valued C^2 function

$$g(s_1, s_2) : \mathbb{R}^2 \rightarrow \mathcal{R}(g) \tag{2}$$

which represents a closed surface. The range $\mathcal{R}(g) \subset \mathcal{W}_{free}$ of the function expresses mathematically the boundary of the surface across which (at a $\delta > 0$ distance) the robot task is to take place. Let us decompose the space around this surface as follows (Fig. 2):

1) The surface's internal, G^-.
2) The surface's boundary, ∂g.
3) The surface's external, G^+.

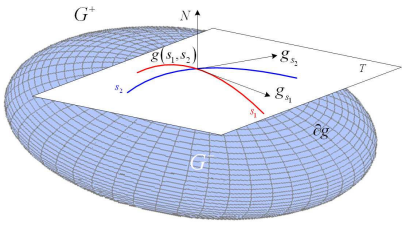

Fig. 2. Representation of tangent's and perpendicular's vectors, of a surface, variations w.r.t parameter's modification.

The tangent vectors on the surface are defined with respect to parameters s_1 and s_2:

$$g_{s_1}(s_1, s_2) = \frac{\partial g(s_1,s_2)}{\partial s_1} = \left[\frac{\partial g_x}{\partial s_1}, \frac{\partial g_y}{\partial s_1}, \frac{\partial g_z}{\partial s_1}\right]^T$$

$$g_{s_2}(s_1, s_2) = \frac{\partial g(s_1,s_2)}{\partial s_2} = \left[\frac{\partial g_x}{\partial s_2}, \frac{\partial g_y}{\partial s_2}, \frac{\partial g_z}{\partial s_2}\right]^T$$

where the g_x, g_y, g_z denote the coordinate functions of g across the respective dimension. Due to the C^2 continuity of $g(s_1, s_2)$, we have that $(g_{s_1} \times g_{s_2}) \neq 0$, $\forall s_1, s_2 \in \mathbb{R}$, [11] , and the vectors g_{s_1}, g_{s_2} are linearly independent everywhere. Therefore, every tangent vector to the surface is a linear combination of the vectors g_{s_1} and g_{s_2}, (Fig. 2). A normalized vector, perpendicular to the surface is then expressed as $N = \frac{g_{s_1} \times g_{s_2}}{\|g_{s_1} \times g_{s_2}\|}$.

The problem is stated as follows: *Given a redundant revolute joint robot manipulator, with kinematic input constraints, operating in a known static and bounded environment, find a feedback kinematic control law that allows the end-effector of the manipulator to*

1) *navigate to any feasible surface point, and*
2) *track a predefined trajectory across the surface.*

III. Controller Design

A. Redundancy Resolution

The system on which our controller will be eventually implemented, is a Mitsubishi PA10-7C, Fig. 3, with $m = 7$ degrees of freedom. The desired position for the manipulator's end-effector in operational space is expressed

$$p_d = \begin{bmatrix} R_d & x_d \\ O^T & 1 \end{bmatrix}$$

and the desired trajectory is represented as and $(p_d(t), \dot{p}_d(t))$, where $R_d \in \mathbb{SO}(3)$ is the rotation matrix, and $x_d \in \mathbb{R}^3$ is the vector of cartesian coordinates of the reference point.

Fig. 3. Mitsubishi PA10-7C robotic manipulator configuration, with 7 d.o.f.

Redundancy is resolved by calculating joint rates that implement the desired trajectory in operational space, while maximizing the manipulator's manipulability. If $J(q)$ is the geometric Jacobian matrix, then the joint rates are expressed in terms of $\dot{p}(t)$ as $\dot{q}_d = J^\dagger \cdot \dot{p}_d + \left(I - J^\dagger \cdot J\right) \dot{q}_a$, [12], where $J^\dagger = J^T \cdot \left(J \cdot J^T\right)^{-1}$ is the right pseudo-inverse of J, and $\dot{q}_a$ is the velocity vector of the redundant degrees of freedom (in the null space of $J(q)$). This vector is chosen so that the manipulability measure $\omega(q) = \sqrt{det(J(q)J^T(q))}$ is maximized. Henceforth, we use the optimal solution for $\dot{q}_d$ to express our reference trajectory on the surface.

B. Workspace Decomposition

The task is completed in two stages, or modes. In the first mode, mode Φ, the end-effector is driven close to the surface. In the second mode, mode $\mathcal{B}$, the robot is steered to a specified point on the surface, or is controlled to track a reference trajectory.

This workspace decomposition requires the definition of a region in which the transition from the one mode to the other occurs. Towards this end, we use the concept of belt zones [4], Fig. 4. The "belt zone" is the region close to the ∂g, consisting of an "internal belt" and an "external belt" region. We assume that the widths of the internal and external belt regions are fixed.

Let us define the vector valued bijective function, [4]

$$a(s_1, s_2) = g(s_1, s_2) + \rho \cdot N(s_1, s_2) \tag{3}$$

with g and N as described in the previous section, and where $0 < \rho < \rho_m$ as in [10]. The vector function that describe the belt zones can be expressed as

$$\beta(s_1, s_2) = g(s_1, s_2) + \delta \cdot N \tag{4}$$

$$\gamma(s_1, s_2) = \beta(s_1, s_2) + \delta \cdot N \tag{5}$$

with $0 < 2 \cdot \delta < \rho_m$, from (3). Surface processing tasks require stabilization of the end-effector on the surface $\beta(s_1, s_2)$, defined by (4).

"Internal belt", $\mathcal{I}$, and the "external belt", $\mathcal{E}$ are defined as (Fig. 4)

$$\mathcal{I} = \{q : k(q) = (1 - \lambda) \cdot \beta + \lambda \cdot g, \lambda \in [0, 1]\}$$

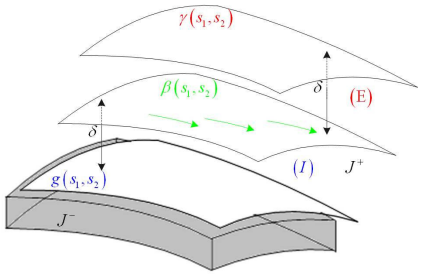

Fig. 4. Representation of Belt Zones, in a part of a surface.

$$\mathcal{E} = \{q : k(q) = (1-\lambda)\cdot\beta + \lambda\cdot\gamma, \lambda \in (0,1]\}$$

Since functions g, β, γ are bijective [10], [9], for every $k(q) \in \mathcal{E} \bigcup \mathcal{I}$ there is a unique couple (s_1, s_2).

C. Navigation Function

The controller's design is based on the navigation function, $\varphi : \mathcal{W} \to \mathbb{R}$, [13]. In the workspace, the volume of the manipulator is represented by a point, using a series of transformations. The obstacles present in the environment are modeled by the navigation function. In order to construct such function, we need to introduce the following parameter $z = q - q_d$, which is the error between the joint configuration of the manipulator, $q \in \mathbb{R}^m$, and the desire joint configuration, $q_d \in \mathbb{R}^m$. The form of the navigation function $\varphi : \mathcal{W}_{ws} \to \mathbb{R}$ according to [4], is as follows:

$$\varphi(z) = \frac{\gamma_d(z)}{\left(\gamma_d^{\kappa}(z) + \beta_{ws}(z)\cdot\beta_{\mathcal{O}}(z)\cdot\beta_s(z)\right)^{\frac{1}{\kappa}}} \qquad (6)$$

where $\gamma_d(z)$ is the distance to goal function,

$$\gamma_d(z) = \gamma_\Phi(z) = \|z\|^2,$$

where the goal is to drive the error to zero, and $\beta_{ws}(z)$ provides the workspace potential, which are defined in dependence on which mode is the end-effector of the manipulator. For mode Φ:

$$\beta_{ws}(z) = \beta_\Phi(z) = -\|q - q_0\|^2 + r_0^2,$$

with $q_0 \in \mathbb{R}^m$ is the joint configuration at the center of the workspace (e.g. the center of the smallest ball containing $\mathcal{W}$), and $r_0 \in \mathbb{R}$ is the workspace's radius. In order to consider the volume occupied by the manipulator, we have used the function $\beta_{\mathcal{O}}(q) \triangleq \prod_{j\in\mathcal{J}}\prod_{i\in\mathcal{I}}\|h^*_{\mathrm{R}_j} - h^*_{\mathcal{O}_i}\|^2$, where $h^*_{\mathrm{R}_j}$ is the position of the transformed robot part j in $\mathcal{W}^*$ and $h^*_{\mathcal{O}_i}$ the position of the transformed obstacle i in $\mathcal{W}^*$, with $\mathcal{W}^*$ is the workspace in which the robot and obstacles are represented by points, as this is defined in [4], and it represents a measure of proximity of the robot to the obstacles. The function $\beta_s(q) \triangleq \prod_{j\in\mathrm{J}}\prod_{k\in\mathcal{K}}\|h^*_{\mathrm{R}_j} - h^*_{s_k}\|^2$ represents the virtual obstacles, in order to achieve singularities avoidance, where $h^*_{s_k}$ are the position of the transformed obstacles, according to [6]. Finally, and $\kappa > 0$ is a parameter.

The convergence to the point on the surface is considered in a two step fashion: First a navigation controller brings the end effector in the belt zone and then a second controller takes over to navigate the system across the surface, and the end-effector is running in the mode Φ. In this mode the surface described by the function $g(s_1, s_2)$, defined in (2) is modeled in the navigation function as an obstacle for the robot's links.

Now it is necessary to define the appropriate form for those function, where the manipulator's end-effector is running in mode $\mathcal{B}$.

To this extend we need to define a navigation function across the 2-D surface, that will provide the navigation vector field. Although theoretically a system that flows according to the tangent space of the 2-D, surface-wrapped navigation field, remains in that 2-D surface, various sources of uncertainty, like sensor noise, model uncertainties and numerical diffusion cause the system to deviate from this surface. To compensate for this problem, we designed an additional vector field perpendicular to the 2-D surface wrapped vector field, which attracts the system on the surface of interest. Such an attractive vector field is provided through an appropriate construction of the γ_d function and by introduction of an additional "perpendicular" workspace function that prohibits exiting the belt zone.

Assume that $h(q)$ is the distance from the surface $g(s_1, s_2)$ on the belt zones. For $h_0 = 0$ we have that the end-effector is on the surface defined by g (boundary of internal region), and for $h_{ext} = 2\cdot\delta$ we have that the end-effector is on the surface defined by γ (boundary of external region). Also, the desired distance from the surface $g(s_1, s_2)$ is at $h_d = \delta$, that is, when the end-effector is in the surface $\beta(s_1, s_2)$. Thus, we can define the distance to the goal function as:

$$\gamma_d(z) = \gamma_{\mathcal{B}}(z) = \left\| \begin{bmatrix} q \\ h(q) \end{bmatrix} - \begin{bmatrix} q_d \\ h_d \end{bmatrix} \right\|^2,$$

where the second term of the vectors is used to attract the end-effector to the surface β. Also the "perpendicular" workspace function is given from the equation:

$$\beta_{ws}(z) = \beta_{\mathcal{B}}(z) = \frac{(h_{ext} - h_d)^2 - (h(q) - h_d)^2}{(h_{ext} - h_d)^2}$$

The function $\beta_{\mathcal{B}}(z)$ in this case guarantees that the robot's end-effector cannot leave the belt zone, (Fig. 5). The workspace boundary for the navigation task is thus defined in both the 2-D workspace, where we usually place it to cover a "bad" region (incorporated in the $\beta_{\mathcal{O}}$ function) and in the "perpendicular" direction to prohibit exiting the belt zone.

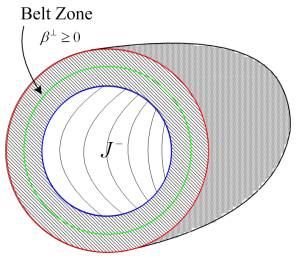

Fig. 5. Representation of the workspace obstacle function based on the Belt Zones construction.

When the manipulator's end-effector is running in the mode $\mathcal{B}$, it can perform a trajectory tracking task. In this case, we have to define an appropriate vector field in order to track the predefined trajectory $(\dot{q}_d,\ q_d)$.

Our system now is time varying, and we have to recast the navigation function as follows:

$$\varphi_{tr}(z,t) = \frac{\gamma_{\mathcal{B}}^{\kappa+1}(z)}{[\gamma_{\mathcal{B}}^{\kappa}(z) + \beta_{\mathcal{B}}(z,q_d(t)) \cdot \beta_{\mathcal{O}}(z,q_d(t)) \cdot \beta_s(z,q_d(t))]^{1/\kappa}} \tag{7}$$

D. Controller Synthesis

Assume that the robot's initial configuration is $q(0) \in \mathbb{R}^m$, with $p(0) = k(q(0)) \in G^+$, and we would like the end-effector move towards the surface, in order to reach a specific point on it.

We will consider the system as operating in two possible modes: **mode Φ** where $p \in \mathcal{W}_{ext}$, where $\mathcal{W}_{ext} = \{\mathcal{W}_{free} \bigcap G^+\} \setminus \{\mathcal{E} \bigcup \mathcal{I}\}$, with $p = k(q)$ the manipulator's direct kinematics, and **mode $\mathcal{B}$** where $p \in \mathcal{E} \bigcup \mathcal{I}$. We define the following vector fields for each mode:

$$\begin{aligned} f_\Phi(z) &= -k_1 \cdot \nabla\varphi_{|\beta_{ws}:=\beta_\Phi,\gamma_d:=\gamma_\Phi}(z) \triangleq -k_1 \cdot \nabla\varphi_\Phi(z) \\ f_{\mathcal{B}}(z) &= -k_2 \cdot \nabla\varphi_{|\beta_{ws}:=\beta_{\mathcal{B}},\gamma_d:=\gamma_{\mathcal{B}}}(z) \triangleq -k_2 \cdot \nabla\varphi_{\mathcal{B}}(z) \end{aligned} \tag{8}$$

where $k_1 > 0$, $k_2 > 0$ are constant parameters. During tracking phase, it holds that:

$$f_{tr}(z,t) \triangleq -k_3 \cdot \nabla\varphi_{\mathcal{B}}(z,t) - \frac{1}{m} \cdot \begin{bmatrix} |\frac{\partial\varphi_{\mathcal{B}}}{\partial z_1}|^{-1} \\ |\frac{\partial\varphi_{\mathcal{B}}}{\partial z_2}|^{-1} \\ \cdots \\ |\frac{\partial\varphi_{\mathcal{B}}}{\partial z_m}|^{-1} \end{bmatrix} \cdot \frac{\partial\varphi_{\mathcal{B}}}{\partial t} \tag{9}$$

where $k_3 > 0$ is constant parameter.

In order to compensate the manipulator's kinematics input constraints, we have to construct an appropriate controller for each of the above modes. We have introduced the following vector field's form:

$$f_i^{new} = sat_{u_{max}}(f_i) \tag{10}$$

with $sat_c(x) = \begin{cases} x & , \quad |x| \le c \\ -c & , \quad x < -c \\ c & , \quad x > c \end{cases}$ where c is a constant, u_{max} is the vector of maximum joint velocity values, and $i = \{\Phi, \mathcal{B}\}$ for each mode of operation, respectively. During tracking phase, we have the following form of the vector field:

$$f_{tr}^{new} = sat_\mu(f_{tr}) \tag{11}$$

where $\mu = u_{max} - \dot{q}_d(t)$, in order to satisfy the velocity input constraints.

IV. Stability Analysis

A. Outside the Belt Zone (*Mode* Φ)

***Proposition** 1:* Consider the system $\dot{z} = v$, where $z = q - q_d$, and $v = u - \dot{q}_d$, with u the control law of (1). This system under the control law $v = f_\Phi^{new}(z)$, with f_Φ^{new} as is defined in (10), is globally asymptotically stable, almost everywhere[1].

Proof:

We use the navigation function $V(z) \triangleq \varphi_\Phi(z)$, (6) as a Lyapunov function candidate. To examine its derivative, we assume ζ_1 the number of f_Φ components that are not saturated, and ζ_2 the saturated components, thus $\zeta_1 + \zeta_2 = m$, and we have that

$$\dot{V} = \nabla V^T \cdot f_\Phi^{new} =$$
$$\sum_{l_1=1}^{\zeta_1} \left(-k_1 \cdot \left| \frac{\partial V}{\partial z_{l_1}} \right|^2 \right) + \sum_{l_2=1}^{\zeta_2} \left[-\text{sign}\left(\frac{\partial V}{\partial z_{l_2}} \right) \cdot \frac{\partial V}{\partial z_{l_2}} \cdot u_{\max}^{l_2} \right] =$$
$$- \sum_{l_1=1}^{\zeta_1} \left(k_1 \cdot \left| \frac{\partial V}{\partial z_{l_1}} \right|^2 \right) - \sum_{l_2=1}^{\zeta_2} \left(\left| \frac{\partial V}{\partial z_{l_2}} \right| \cdot u_{\max}^{l_2} \right),$$

where $k_1 > 0$. Thus, $\dot{V}$ is strictly negative unless $\nabla V = \nabla\varphi_\Phi = 0$. Since φ_Φ is a navigation function [13], the condition $\nabla\varphi_\Phi = 0$ holds only at the destination configuration, and a set of isolated saddle points. By construction [13], the region of attraction of these saddle points is a set of measure zero. Thus, the system converges to the destination configuration from almost everywhere. ■

B. Inside the Belt Zone (*Mode* $\mathcal{B}$)

Since the robot's end-effector initial condition is $p_0 \in G^+$ and by construction it holds that $p_d \in G^-$, the solutions of (1), which are absolutely continuous, intersect the surface $\gamma(s_1, s_2)$, using standard topological arguments, [14]. Therefore there exists finite time T for which the system enters the belt zones. When in the belt zone a mode switch occurs that activates mode $\mathcal{B}$. Once the robot end-effector enters the belt zone, it remains there as the boundaries of the belt zone are repulsive due to the construction of the workspace. Therefore, it has to execute the stabilization over the surface task, and the trajectory tracking task.

1) Stabilization on the Surface:

***Proposition** 2:* The system $\dot{z} = v$ under the control law $v = f_{\mathcal{B}}^{new}(z)$, with $f_{\mathcal{B}}^{new}$ as is defined in (8), is globally asymptotically stable, a.e.

Proof: The proof is very similar to that of Proposition 1, using $V(z) \triangleq \varphi_{\mathcal{B}}(z)$ as a Lyapunov function candidate. ■

2) Tracking on the surface:

***Proposition** 3:* The solutions of system $\dot{z} = v$ under the control law $v = f_{tr}^{new}(z,t)$, with f_{tr}^{new} as is defined in (11), are globally uniformly ultimately bounded with the bound that can be made arbitrarily small with the suitable choice of k_3.

Proof: Using the same notation as in the previous proofs, we define a time-varying, continuously differentiable Lyapunov function candidate $V(z,t) \triangleq \varphi_{tr}(z,t)$, (7).

Inheriting the properties of navigation functions, function $V(z,t)$ is positive definite for all $t > 0$, and there is therefore a class $\mathcal{K}$ function $V_1(\|z\|)$ which satisfy $0 < V_1(\|z\|) \le V(z,t)$. In addition, there exists a sphere centered at the the

[1] i.e. everywhere except a set of initial conditions of measure zero.

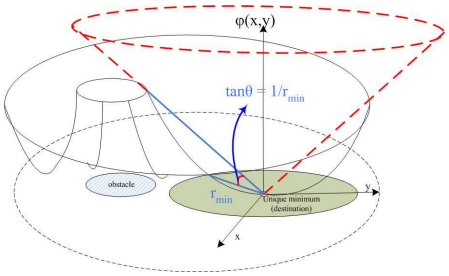

Fig. 6. A navigation function over a free configuration space. It is easy to find the minimum radius r_{min} (radius of a tangent to an obstacle sphere, centered at the destination point), which guarantee that the destination point is collision free.

destination point $q = q_d(t)$ (i.e., $z = 0$), with radius $r(t)$, which is tangent to, but not intersecting with, an obstacle in the workspace. For a constant $r_{min} \triangleq \inf_{t>0} r(t)$, we can define a cone $V_2(\|z\|) = c_1 \cdot \|z\|$ with $c_1 = \frac{1}{r_{min}}$, such that $V(z,t) \leq V_2(\|z\|)$ (Fig. 6). Thus, $V_1(\|z\|) \leq V(z,t) \leq V_2(\|z\|)$.

Assume ξ_1 the number of f_{tr} components that are not saturated, and ξ_2 the saturated components, thus $\xi_1+\xi_2 = m$. Then,

$$\dot{V} = \frac{\partial V}{\partial t} + \nabla V^T \cdot f_{tr}^{new} =$$
$$\frac{\partial V}{\partial t} + \sum_{l_1=1}^{\xi_1} \left(-k_3 \cdot \left| \frac{\partial V}{\partial z_{l_1}} \right|^2 - \frac{1}{m} \cdot \frac{\partial V}{\partial t} \right) +$$
$$\sum_{l_2=1}^{\xi_2} \left[-\text{sign}\left(\frac{\partial V}{\partial z_{l_2}} \right) \cdot \frac{\partial V}{\partial z_{l_2}} \cdot \mu_{l_2} \right] =$$
$$\frac{\partial V}{\partial t} - \sum_{l_1=1}^{\xi_1} \left(k_3 \cdot \left| \frac{\partial V}{\partial z_{l_1}} \right|^2 + \frac{1}{m} \cdot \frac{\partial V}{\partial t} \right) - \sum_{l_2=1}^{\xi_2} \left(\left| \frac{\partial V}{\partial z_{l_2}} \right| \cdot \mu_{l_2} \right)$$

The first term in this equation can be written in the form $\frac{\partial V}{\partial t} = -\frac{\gamma_d^{\kappa+1}}{\kappa \cdot \left(\gamma_d^k + Q\right)^{\frac{\kappa+1}{\kappa}}} \cdot \frac{\partial Q}{\partial t}$, where $Q = (\beta_{\mathcal{B}} \cdot \beta_{\mathcal{O}} \cdot \beta_s)$. It can therefore be bounded as follows:

$$\left| \frac{\partial V}{\partial t} \right| < \frac{1}{\kappa} \cdot \gamma_d^{\kappa+1} \cdot \frac{1}{\left(\gamma_d^{\kappa}\right)^{\frac{\kappa+1}{\kappa}}} \cdot \left| \frac{\partial Q}{\partial t} \right| = \frac{1}{\kappa} \left| \frac{\partial Q}{\partial t} \right|$$

The properties of the navigation function ensure boundedness of its gradient within the workspace, and therefore ensure the existence of a positive bound $\theta = \left(\max_{\bar{\mathcal{W}}/\mathcal{M}} \left\| \frac{\partial V}{\partial z_{l_2}} \right\|_{\infty} \right)^{-1}$, where $\mathcal{M}$ is a set of measure zero, including the unstable saddle points of the navigation function as well as the destination configuration. We can then write $\theta \cdot \left| \frac{\partial V}{\partial z_{l_2}} \right| < \mu_{l_2}$, $\forall l_2$, and bound the last term in $\dot{V}$ as

$$-\sum_{l_2=1}^{\xi_2} \left(\left| \frac{\partial V}{\partial z_{l_2}} \right| \cdot \mu_{l_2} \right) < -\sum_{l_2=1}^{\xi_2} \left(\left| \frac{\partial V}{\partial z_{l_2}} \right|^2 \cdot \theta \right)$$

Thus, it holds that

$$\dot{V} \leq \left(1 - \tfrac{\xi_1}{m}\right) \cdot \frac{\partial V}{\partial t} - \theta \cdot \|\nabla V\|^2 - \sum_{l_1=1}^{\xi_1} \left(k_4 \cdot \left| \frac{\partial V}{\partial z_{l_1}} \right|^2 \right),$$

where the control gain k_3 can be then decomposed in the form $k_3 = \theta + k_4$, with $k_4 > 0$. Then,

$$\dot{V} \leq (1 - \tfrac{\xi_1}{m}) \tfrac{1}{\kappa} \sup_{\bar{\mathcal{W}}} \left| \frac{\partial Q}{\partial t} \right| - \theta \cdot \|\nabla V\|^2,$$

in which $\sup_{\bar{\mathcal{W}}} \left| \frac{\partial Q}{\partial t} \right|$ depends on $\sup_{\bar{\mathcal{W}}} \|\dot{q}_d\|$. In the region where

$$\|\nabla V\| > \sqrt{\frac{\left(1 - \frac{\xi_1}{m}\right) \frac{1}{\kappa} \sup_{\bar{\mathcal{W}}} \left| \frac{\partial Q}{\partial t} \right|}{\theta}}$$

the Lyapunov function is decreasing, and therefore, z converges to 0 which corresponds to the destination configuration ($q_d(t)$). Thus, z is uniformly ultimately bounded in the region where the above condition holds.

In the neighborhood of $z = 0$, ∇V does not vanish except for $z = 0$, since V is defined to be equal to a navigation function of z. Thus, $\|\nabla V\|$ is a positive definite scalar function, and thus there exist $V_3(\|z\|)$, $V_4(\|z\|)$ class $\mathcal{K}$ functions for which $V_3(\|z\|) \leq \|\nabla V\| \leq V_4(\|z\|)$. Using the lower bounding function V_3, if

$$\|z\| \geq V_3(\|z\|)^{-1} \cdot \sqrt{\frac{\left(1 - \frac{\xi_1}{m}\right) \frac{1}{\kappa} \sup_{\bar{\mathcal{W}}} \left| \frac{\partial Q}{\partial t} \right|}{\theta}} \in \mathcal{K}, \quad (12)$$

then, for the gradient of V we can write

$$\|\nabla V\| \geq V_3(\|z\|) \geq \sqrt{\frac{\left(1 - \frac{\xi_1}{m}\right) \frac{1}{\kappa} \sup_{\bar{\mathcal{W}}} \left| \frac{\partial Q}{\partial t} \right|}{\theta}},$$

implying that $\dot{V}$ is strictly negative in the region defined by (12).

Application of Theorem 1 [14] (see Appendix), ensures that z is globally uniformly ultimately bounded. ■

Remark 1: There are two ways in which we can turn uniform ultimate boundedness to uniform asymptotic stability in Proposition 3. When $\dot{q}_d \to 0$, the region defined by (12) reduces to a ball of radius zero, because as $\dot{q}_d \to 0$, we have $\frac{\partial Q}{\partial t} \to 0$. Another way is to ensure that no component of f_{tr} is saturated; then $\xi_2 = 0 \Rightarrow \xi_1 = m$, and the right hand side of (12) vanishes.

C. Combined Hybrid System

The transition from mode Φ to mode $\mathcal{B}$ is happening just once, and then the system is staying at the second mode, since the belt zone's workspace is positively invariant. According to *Theorem* 2 (Appendix), it holds that, the switched system is globally asymptotically stable a.e., since the condition of this theorem

$$V_{\mathcal{B}}(z,t) - V_{\Phi}(z,t_1) \leq -W$$

is trivially satisfied, because we have just one switch from the mode Φ to the mode $\mathcal{B}$, with V_ϕ and $V_{\mathcal{B}}$ the Lyapunov functions for each mode, t_1 is the time in which the switch from the mode Φ to the mode $\mathcal{B}$ is occurred, and W is a positive definite continuous function.

V. Simulation Results

Computer simulations have been carried out to verify the feasibility and efficacy of the proposed methodology. The robot manipulator that we have used for the implementation of the simulations, is the model of Mitsubishi PA10-7C, in the configuration of Fig. 3, with $m = 7$ d.o.f. The

vector of joint's velocity limitation in (rad/sec), is $u_{max} = \begin{bmatrix} 0.6 & 0.6 & 1.2 & 1.2 & 1.2 & 1.2\pi & 1.2\pi \end{bmatrix}^T$. The scenario of the simulation contains two 3-D (ellipsoid) obstacles centered at $\mathcal{O}_1 : (-0.3, -0.4, 0.1)$ and $\mathcal{O}_2 : (0.35, -0.3, -0.5)$, and with semi-axes lengths $(0.05, 0.10, 0.20)$, both of them. The surface of interest $g(s_1, s_2)$ is assumed to be an ellipsoid, centered at $(0, 0, 0)$ with semi-axes lengths $(0.75, 0.25, 0.35)$.

The "bad" region's obstacles, which are the areas on the surface that the robot cannot approach, are centered at $\mathcal{O}_{g1} : (-0.33, -0.08, 0.18)$, $\mathcal{O}_{g2} : (0.33, -0.08, -0.18)$ and $\mathcal{O}_{g3} : (-0.33, -0.08, -0.18)$. The robot manipulator's initial configuration was $p(0) = (-0.61, -0.39, -0.13, 0.0, 0.0, 0.0)$, and the target configuration in the operational space was set at $p_d = (0.49, -0.16, 0.13, 1.33, 0.87, -1.33)$. After the end-effector of the robot manipulator reaches its destination point (p_d) (the end-effectors path is red line), it starts a tracking task (blue line), to track a predefined trajectory which is the yellow colored line in Fig. 7.

Fig. 7 present the simulation results from a different point of view. Fig. 8, 9 present the cartesian position and the error between the real cartesian position and the desired position during tracking, while Fig. 10, 11 depict the joint's position and velocity, respectively. The time axis in those figures is refered to both tasks (reaching a point and then trajectory tracking), while in Fig. 9 the depicted time (x axis) is refered to trajectory tracking task only. Our algorithm successfully converges to the goal configuration and track the predefined trajectory avoiding obstacles.

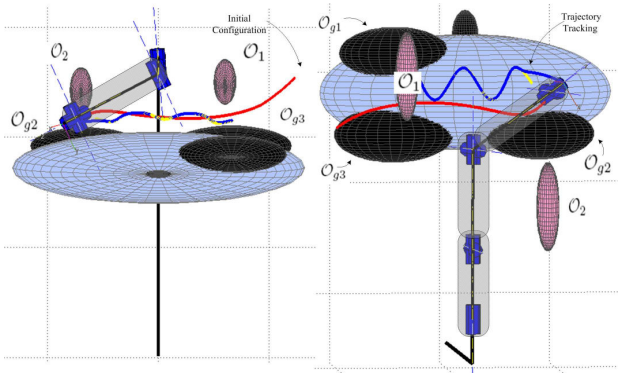

Fig. 7. Reaching a point on the surface and tracking.

VI. Conclusion and Future Work

We presented a methodology for performing navigation and tracking tasks over a 2-dimensional manifold embedded in a 3-dimensional workspace applicable to articulated robotic manipulators, with kinematic input constraints. After safely navigating the manipulator's end-effector to the 2-D manifold, task specific vector fields direct the end-effector towards accomplishing a navigation or a trajectory tracking task across the 2-D manifold. The methodology has theoretically guaranteed global convergence and collision avoidance properties.

We can easily apply this methodology to dynamical control for a redundant manipulator while it performs surface

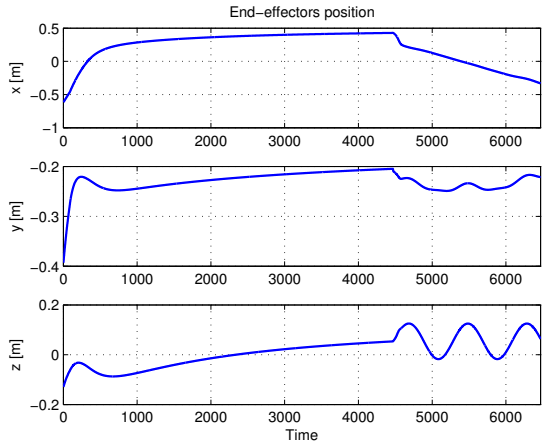

Fig. 8. Robot end-effector cartesian position, where the time axis is refered to both tasks of reaching a point on the surface and trajectory tracking task).

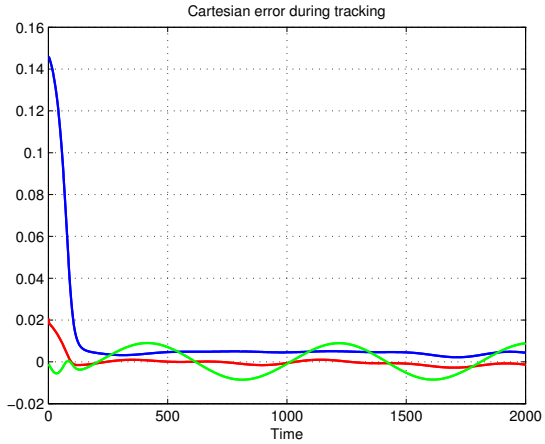

Fig. 9. Cartesian position error between the real position of the robot and the predefined trajectory during trajectory tracking, blue - x, red - y, green - z.

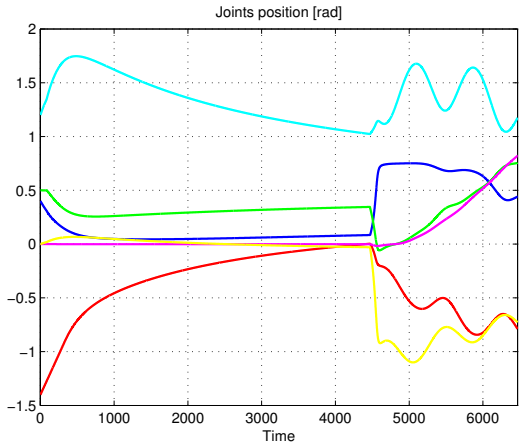

Fig. 10. Joint's angles, blue - q_1, red - q_2, green - q_3, cyan - q_4, magenta - q_5, yellow - q_6, black - q_7.

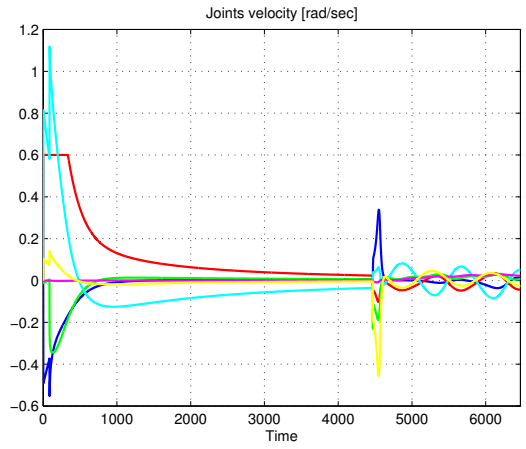

Fig. 11. Joint's velocities, blue - q_1, red - q_2, green - q_3, cyan - q_4, magenta - q_5, yellow - q_6, black - q_7.

tasks. This dynamic control approach is using the back-stepping methodology by using the established closed loop control for the kinematic subsystem (1), which is described in this work.

Further research includes considering surface properties in the construction of the belt zone vector fields and implementing the methodology to real neuro-robotic systems taking into account their dynamics and kinematic constraints.

APPENDIX

Theorem 1: [14] Let $\mathcal{D} \subset \mathbb{R}^n$ be a domain that contains the origin and $V : [0, \infty) \times \mathcal{D} \rightarrow \mathbb{R}$ be a continuously differentiable function such that

$$W_1(x) \leq V(t,x) \leq W_2(x)$$
$$\frac{\partial V}{\partial t} + \frac{\partial V}{\partial x} f(t,x) \leq -W_3(x) \ , \ \forall \|x\| \geq \lambda > 0$$

$\forall t \geq 0$, $\forall x \in \mathcal{D}$ where $W_1(x)$ and $W_2(x)$ are class $\mathcal{K}$ functions and $W_3(x)$ is a continuous positive definite functions. Take $r > 0$ such that $\mathcal{B}_r \subset \mathcal{D}$ and suppose that

$$\lambda < W_2^{-1}\left(W_1(r)\right)$$

Then, there exist a class $\mathcal{KL}$ function b for every initial state $x(t_0)$, satisfying $\|x(t_0)\| \leq W_2^{-1}\left(W_1(r)\right)$, there is $T \geq 0$ (dependent on $x(t_0)$ and λ) such that $\forall x(t_0) \in \{x \in \mathcal{B}_r | W_2(x) \leq \rho\}$, the solution of $\dot{x} = f(t,x)$ satisfies

$$\|x(t)\| \leq b\left(\|x(t_0)\|, t - t_0\right) \quad , \ \forall t_0 \leq t \leq t_0 + T$$
$$\|x(t)\| \leq W_1^{-1}\left(W_2(\lambda)\right) \quad , \ \forall t \geq t_0 + T$$

Moreover, if $\mathcal{D} = \mathbb{R}^n$ and W_1 belongs to class $\mathcal{K}_\infty$, then the last two conditions hold for any initial state $x(t_0)$, with no restriction on how large λ is.

Theorem 2: [15] Let $\dot{x} = f_\upsilon(x)$, be a finite family of globally asymptotically stable systems, and let V_υ, $\upsilon \in \mathcal{P}$ be a family of corresponding radially unbounded Lyapunov functions, where $\mathcal{P}$ is some index set. Suppose that there exists a family of positive definite continuous functions W_υ, $\upsilon \in \mathcal{P}$, with the property that for every pair of switching times (t_i, t_j), $i < j$ such that $\sigma(t_i) = \sigma(t_j) = \upsilon \in \mathcal{P}$, and $\sigma(t_k) \neq \upsilon$, for $t_i < t_k < t_j$, where σ is the switching signal, we have that

$$V_\upsilon(x(t_j)) - V_\upsilon(x(t_i)) \leq -W_\upsilon(x(t_i))$$

Then the switched system is globally asymptotically stable.

REFERENCES

[1] P. Atkar, D. Conner, A. Greenfield, H. Choset, and A. Rizzi, "Uniform coverage of simple surfaces embedded in $\mathbb{R}^3$ for auto-body painting." Carnegie Mellon University, 2004.

[2] P. Atkar, H. Choset, and A. Rizzi, "Towards optimal coverage of curve," *Proceedings of the IEEE/RSJ Int. Conference on Intelligent Robots and Systems*, 2003.

[3] D. Conner, P. Atkar, A. Rizzi, and H. Choset, "Deposition modeling for paint application on surfaces embedded in $\mathbb{R}^3$," Carnegie Mellon University," Tech. Report, 2002.

[4] X. Papageorgiou, S. Loizou, and K. Kyriakopoulos, "Motion tasks for robot manipulators on embedded 2-D manifolds," *Joint 2006 IEEE Conference on Control Applications (CCA), 2006 IEEE Computer Aided Control Systems Design Symposium (CACSD) & 2006 IEEE International Symposium on Intelligent Control (ISIC)*, pp. 3047–3052, 2006.

[5] D. Koditschek and E. Rimon, "Robot navigation functions on manifolds with boundary," *Advances Appl. Math.*, vol. 11, pp. 412–442, 1990.

[6] H. Tanner, S. Loizou, and K. Kyriakopoulos, "Nonholonomic navigation and control of cooperating mobile manipulators," *IEEE Trans. on Robotics and Automation*, vol. 19, no. 1, pp. 53–64, 2003.

[7] D. Dimarogonas, M. Zavlanos, S. Loizou, and K. Kyriakopoulos, "Decentralized motion control of multiple holonomic agents under input constraints," *42nd IEEE Conference on Decision and Control*, 2003, submitted for Journal Publication.

[8] S. Loizou and K. Kyriakopoulos, "Navigation of multiple input constraint micro-robotic agents," *2005 International Conference on on Intelligent Robots and Systems*, 2005.

[9] S. Loizou, H. Tanner, V. Kumar, and K. Kyriakopoulos, "Closed loop motion planning and control for mobile robots in uncertain environmentss," *Proceedings of the 42nd IEEE Conference on Decision and Control*, 2003.

[10] X. Papageorgiou, S. Loizou, and K. Kyriakopoulos, "Motion planning and trajectory tracking on 2-D manifolds embedded in 3-D workspaces," *2005 IEEE International Conference on Robotics and Automation*, pp. 501–506, 2005.

[11] W. M. Boothby, *An introduction to differentiable manifolds and Riemannian geometry*. Academic Press, 1986.

[12] L. Sciavicco and B. Siciliano, *Modeling and Control of Robot Manipulators*. McGraw-Hill, 1996.

[13] E. Rimon and D. Koditschek, "Exact robot navigation using artificial potential functions," *IEEE Transactions on Robotics and Automation*, vol. 8, no. 5, pp. 501–518, 1992.

[14] H. Khalil, *Nonlinear Systems*. Prentice-Hall, 1996.

[15] D. Liberzon, *Switching in Systems and Control*. Birkhauser, 2003.