Robotica (2012) volume 30, pp. 743–753. © Cambridge University Press 2011
doi:10.1017/S0263574711001019

Recursive field estimation and tracking for autonomous manipulation

Soo Jeon

Department of Mechanical and Mechatronics Engineering, University of Waterloo
200 University Ave. West, Waterloo ON, N2L 3G1, Canada

(Accepted August 11, 2011. First published online: September 15, 2011)

SUMMARY
Autonomous operation of mechanical systems often requires the ability to detect and locate a particular phenomenon occurring in the surrounding environment. Being implemented to articulated manipulation, such a capability may realize a wide range of applications in autonomous maintenance and repair. This paper presents the sensor-driven task space control of an end-effector that combines the field estimation and the target tracking in an unknown spatial field of interest. The radial basis function network is adopted to model spatial distribution of an environmental phenomenon as a scalar field. Their weight parameters are estimated by a recursive least square method using collective measurements from the on-board sensors mounted to the manipulator. Then the asymptotic source tracking has been achieved by the control law based on the gradient of the estimated field. A new singularity tolerant scheme has been suggested to command the task space control law despite singular configurations. Simulation results using the three-link planar robot and the 6-revolute elbow manipulator are presented to validate the main ideas.

KEYWORDS: Control of robotic systems; Automation; Mechatronic systems; Motion planning; Manufacturing.

1. Introduction

Using robotic manipulators for autonomous inspection and repair has been a major concern in a number of industrial sectors,[1] especially in the operation of hazardous or remotely located facilities such as the nuclear reactor[2] and the offshore oil and gas platform.[3] Typically, the basic hardware of manipulators used in such applications is similar to those of conventional ones. Some of different features are that they are often configured on moving platforms for mobility and are usually equipped with a variety of sensors to perform wide range of monitoring and inspection operations.[3] Kinematic redundancy is desired in some cases for extra degrees of freedom to enable dexterous maneuvers in complex environments.[4] However, from the control point of view, the required motion tasks in these applications are quite different from those commonly executed by conventional manipulators. Neither the target point nor the desired trajectory is available beforehand. Instead, the movement of the end-effector is resolved by an instantaneous motion command determined while it navigates through an unknown scalar field of interest. For instance, in the emergency repair of a gas leak in a process plant, the robot needs to command its motion tasks while it traces out the origin of the leak. An interesting problem that arises from such application is the motion control of the end-effector that can achieve both the real-time motion planning and the field estimation simultaneously. The real-time motion planning assumes that the desired motion command can be computed on the basis of field measurement data available up to the present time.

The real-time estimation of an unknown field has recently been an important issue in mobile sensor networks, especially in their application to environmental monitoring and mapping.[5–7] The main concern is how to perform the motion planning of mobile sensor agents while they cooperatively collect measurement data and estimate the spatial field of interest, often in a recursive way. The common approach is to progressively refine the knowledge of the unknown field based on new observations that are available at every instant. There exist several different approaches depending on how we assume the mathematical model of the unknown field and how to design the motion command. Martinez[6] employed the non-parametric inference method where the spatial field is represented by interpolation without *a priori* knowledge on the structure of the unknown field. In contrast, Choi *et al.*[5] assumed that the field can be represented by the weighted sum of a number of radial basis functions (RBF). Leonard *et al.*[7] formulated the gradient estimation scheme considering an affine function as an approximation of the field. In order to trace a target source (e.g., the origin of toxic chemical leak) through the unknown spatial field, the most common approach has been the use of gradient climbing. In fact, it is well known that the gradient climbing is one of the most natural behaviors of biological species in searching for their target resources such as the densest source of food or the most favorable habitats.[8] These observations have recently motivated various applications of mobile robot platforms such as the environmental monitoring[5] and the instantaneous source localization[9] of toxic gas release. It should be noted that the motion control of the end-effector with local gradient has long been studied under the framework of the artificial potential field, which was recognized as a unified approach to combine the target tracking with collision avoidance.[10]

By combining the on-line field estimation with the task space control using artificial potential field, this paper

* Corresponding author. E-mail: soojeon@uwaterloo.ca

https://doi.org/10.1017/S0263574711001019 Published online by Cambridge University Press

presents a new sensor-driven control strategy for control of the robot end-effector in an unknown field. The work in this paper is distinct from existing results because it combined two techniques that have been developed separately: articulated manipulation with potential field and the sensor-based adaptive target tracking. A critical issue arising from the implementation of the suggested control scheme is how to handle the kinematic singularity that may be encountered during the navigation. Existing singularity robust algorithms such as the damped least square (DLS)[11] or the adjoint Jacobian method[12] along with more recent ones[13] are mainly effective for a known trajectory only. In case of artificial potential field approach, where the trajectory is not defined beforehand, an additional potential function can be added to avoid the singularity by restraining the joint angle[14] or using the manipulability measure.[15] If the manipulator is redundant, the gradient of these additional potential functions can be augmented by the projection onto the null space of the Jacobian to keep the joint angles away from the singular configurations without compromising the task space control action.[16] While this method can prevent the redundant manipulator from entering singularities by taking advantage of its self-motion, a problem still occurs if the task space motion command from the field estimation is persistently directing the end-effector toward the boundary of the workspace. To overcome such issues, this paper proposes a new control strategy that can still drive the end-effector as close as possible to the target source while tolerating the singular configuration at or near the boundary of the work space.

This paper is organized as follows. Section 2 presents the problem statement and basic formulations. Then the main results follow in Section 3, which describes the formulation of the recursive field estimation and the design of the singularity tolerant motion command, as well as a brief convergence analysis. Simulation results with a three-link planar manipulator and a 6-revolute (6R) elbow manipulator are given in Section 4 to verify the proposed scheme. The concluding remarks are presented in Section 5.

2. Problem Statement and Basic Formulations

Consider that a robot manipulator is deployed to a certain region where the unknown field of interest is distributed over the operational range of the manipulator. Without *a priori* knowledge on the location of the target source, the manipulator needs to run a learning process to make its way through the field to find the target source. As the sensors collect more data about the field, the manipulator will continuously build a better knowledge on the configuration of the field as well as its gradient vector toward which the motion command is being applied. Eventually, it needs to converge to the target source, i.e., the global maximum of the field. Such a concept is illustrated in Figure 1, where the distribution of the unknown field is represented by the iso-surfaces and their gradient vectors. The manipulator will be instrumented with a number of field sensors at different parts of the body to provide enough data to estimate the parameters characterizing the field and also to identify various objects

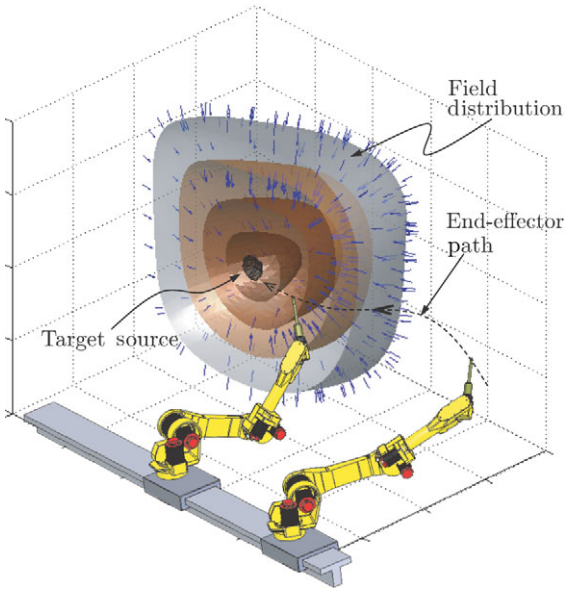


Fig. 1. (Colour online) A robot manipulator and an unknown field of interest.

to be avoided during navigation. The main objective of this paper is to formulate a task space control strategy to enable the end-effector of the manipulator continuously directed toward the target source of the unknown field. In this paper, the term field refers to a scalar field representing a spatial distribution of an environmental phenomenon. It is assumed that the scalar value at each point is the only measurement available through local sensing. It excludes the physical environment that can be recognized by cognitive sensing such as vision, ultrasonic, or haptic devices. Temperature distribution, diffusion of chemicals or magnetic/electric field can be taken as examples of scalar field. For simplicity, we will focus on the static field in this paper but the results can also be easily extended to the dynamic field (i.e., a spatiotemporal field).

In the remaining, we will assume that the spatial field has a unique maximum point or equivalently, the gradient of the potential field does not vanish except for the unique global maximum. Moreover, since we are concerned about the source localization, the task space variables of the end-effector will be constrained to the Cartesian coordinates only, i.e., the orientation of the end-effector will be regarded as redundancy in degrees of freedom. This is because the scalar field sensors considered in this paper are assumed to be able to detect only the magnitude of the field, thus making the target expressed by a point in the space.

2.1. Manipulator dynamics in task space

Consider a mechanical manipulator with n links, the dynamics of which is given by the well-known Euler-Lagrange equation as[17]

$$M(q)\ddot{q} + L(q, \dot{q})\dot{q} + G(q) + F_f(\dot{q}) = \tau + F_d, \quad (1)$$

where $M(q) \in \mathbb{R}^{n\times n}$ is the inertia matrix, which is symmetric and positive definite, $L(q, \dot{q})\dot{q} \in \mathbb{R}^n$ is the Coriolis and

https://doi.org/10.1017/S0263574711001019 Published online by Cambridge University Press

centrifugal force vector, $G(q) \in \mathbb{R}^n$ is the gravitational force vector, $F_f(\dot{q}) \in \mathbb{R}^n$ is the nonlinear joint friction force vector, $F_d \in \mathbb{R}^n$ is the vector of unknown external disturbance forces in actuators, and $q(t)$, $\dot{q}(t)$, and $\ddot{q}(t) \in \mathbb{R}^n$ are vectors for joint positions, joint velocities, and joint accelerations, respectively. We will assume that the manipulator is reasonably rigid considering it moves with relatively slow speed carrying a small or no payload. Let $x(t) \in \mathbb{R}^\ell$ represents the configuration of the end-effector in the task space. Note that $\ell = n$ if we assume the fully actuated and non-redundant manipulator. $x(t)$ is related to the joint space vector $q(t)$ by[17]

$$\begin{aligned} &x(t) = \mathcal{K}(q(t)), \quad \dot{x}(t) = v(t) = J(q(t))\dot{q}(t), \\ &\ddot{x}(t) = a(t) = J(q(t))\ddot{q}(t) + \dot{J}(q(t))\dot{q}(t), \end{aligned} \tag{2}$$

where $\mathcal{K} : \mathbb{R}^n \to \mathbb{R}^\ell$ is the forward kinematics that maps the joint space variable $q(t)$ to the task space variable $x(t)$ and $J(q) = \frac{\partial f}{\partial q} \in \mathbb{R}^{\ell \times n}$ is the Jacobian manipulator. For simplicity, the arguments of the functions will be omitted in the remaining except when needed to avoid any confusion. Consider a modified inverse dynamics control law of the form[17]

$$\tau = \hat{M} a_q + \hat{L}\dot{q} + \hat{G} + \hat{F}_f - \hat{F}_d, \tag{3}$$

where $\hat{M}$, $\hat{L}$, $\hat{G}$, $\hat{F}_f$, and $\hat{F}_d$ denote the estimated (computed) values for the corresponding time-varying parameters and a_q is an equivalent desired acceleration referenced to the joint space coordinate system. From Eq. (2), we can map the desired joint acceleration a_q to the desired task space acceleration a_x as[17]

$$a_x = \hat{J} a_q + \hat{\dot{J}}\dot{q}, \tag{4}$$

where $\hat{J}$ and $\hat{\dot{J}}$ are the estimated values of the Jacobian and its derivative, respectively. The hat notation ($\hat{\bullet}$) will be used as the estimated value for the corresponding variable throughout the paper. Substituting Eqs. (3) and (4) to Eq. (1), we get the decoupled dynamic system in task space as

$$\dot{v} = a_x + \mathcal{F}_p, \tag{5}$$

where $\mathcal{F}_p$ is a small perturbation term given by

$$\begin{aligned} \mathcal{F}_p = JM^{-1}\{&\Delta(MJ^{-1})a_x + \Delta(L - MJ^{-1}\dot{J})\dot{q} \\ &+ \Delta(G + F_f - F_d)\}, \end{aligned} \tag{6}$$

where Δ denotes the parameter error for each term, i.e., $\Delta(\bullet) := (\hat{\bullet}) - (\bullet)$. In general, high performance manipulator control using Eq. (3) will require accurate knowledge of robot models. This is the case not only for conventional industrial robots but also for more advanced ones. In practice, additional care should be exercised in implementing the decoupling control law based on Eq. (3). Variations of the task space control for practical implementation can be found in the recent literature.[21,22]

Assuming the desired task space acceleration a_x and the joint velocity $\dot{q}$ are bounded, and also the parameter errors in Eqs. (3) and (4) are small, we can consider $\mathcal{F}$ bounded by a maximum value, say $\mathcal{F}_p^{\max}$ at all times, i.e.,

$$\|\mathcal{F}_p(t)\| \leq \mathcal{F}_p^{max}, \quad \forall t \geq 0. \tag{7}$$

In this paper, the desired task space acceleration a_x will take the following form:

$$a_x(t) = \Gamma_f - K_v \hat{v}(t), \tag{8}$$

where Γ_f is the field tracking control law, which will be derived in Section 3.2. The second term on the right-hand side is the damping effect realized by the product of the estimated task space velocity, $\hat{v}(t) = \hat{J}\dot{q}(t)$. The diagonal matrix $K_v \in \mathbb{R}^{\ell \times \ell}$ represents damping coefficients. Note that it represents the task space damping which will be realized through the modified inverse dynamics control law given by Eq. (3).

2.2. *Redundancy resolution*

As previously mentioned, it is often desired to have redundancy in the link mechanism when the manipulator is expected to navigate through a complex environment. In such cases, the degree of freedom (ℓ) in the task space is smaller to the number of links (n). Since the Jacobian matrix possesses more columns than its rows, the joint space motion variables cannot be uniquely determined by a set of parameters that only describe the end-effector motion in task space. In this case, we have extra degrees of freedom in designing the joint control force to realize a desired task space motion. Note that our control task does not presume any knowledge on the desired trajectory, so we are mostly concerned about redundancy resolution at the acceleration level. There are several methodologies to resolve the redundancy. The typical approach is to take advantage of the extra degrees of freedom in joint space to satisfy different performance objectives such as torque and acceleration minimization, singularity, or obstacle avoidance.[19] From the perspective of unknown field tracking, it is crucial to provide a means to keep the robot from the singular configurations at every instance.

Equation (4) still holds in the redundant case, but a_q cannot be uniquely determined since $\hat{J}$ is not invertible. The general solution in this case can be given by[15]

$$a_q = \hat{J}^\dagger\left(a_x - \hat{\dot{J}}\dot{q}\right) + \left(I - \hat{J}^\dagger\hat{J}\right)\eta, \tag{9}$$

where $\hat{J}^\dagger$ is the pseudo-inverse of $\hat{J}$ defined as

$$\hat{J}^\dagger := \hat{J}^T\left(\hat{J}\hat{J}^T\right)^{-1}, \tag{10}$$

and $\eta \in \mathbb{R}^n$ is an arbitrary vector for the extra degrees of freedom. Singularity avoidance can be implemented in several ways. One approach is to use a potential (or objective)

https://doi.org/10.1017/S0263574711001019 Published online by Cambridge University Press

function and apply its gradient as an internal force, i.e.,

$$\eta = -k_s \nabla V_s(q) = -k_s \frac{\partial V_s(q)}{\partial q}, \tag{11}$$

where k_s is a constant coefficient for step size and V_s is the objective function representing the measure of singularity. We may choose V_s as the manipulability measure,[15] which can be written as

$$V_s = W := -\det\left(\hat{J}\hat{J}^T\right). \tag{12}$$

Thus, being mapped to the null space of $\hat{J}$, the additional joint torque η keeps the joint positions away from the singular configurations without changing the task space motion command applied by a_x. Note that this approach of singularity avoidance is valid only when the manipulator is operating strictly within its operation range whereby the redundancy is preserved. When it approaches the boundary of operation range, it is inevitable to encounter singular configuration. This problem will be handled in Section 3.3.

3. Spatial Field Estimation and Tracking

In practice, the joint feedback control law given by Eq. (3) is implemented in a digital controller. Hence, it is reasonable to run the joint control loop at the sample rate fast enough to realize the desired motion control performance. On the other hand, the control action that will be given by the estimated field data (which will be implemented to a_x in Eq. (5) by a recursive least square (RLS) method) does not need to run at such a fast rate. As will be clear in later sections, there are two reasons for this. Firstly, the RLS algorithm updates parameters based on new observations from the field sensors. These new measurements may not contain new information until the end-effector moves to a new location where the data are distinguishable from the previous measurement. Considering the usual speed of an end-effector, updating the measurement at every base sample rate may not be needed. Secondly, the computation load of the RLS scheme increases by the square of the number of basis functions as well as that of the field sensors used. So if we want to include enough number of basis functions (or to use more sensors) to cover a wide range of field configuration, the computation load may become quite intensive, which will slower the achievable sample rate for the RLS routine. Hence, in the remaining of the paper, we will formulate the RLS in discrete time domain to account for different time scale of the RLS compared to other motion control signals.

3.1. Recursive estimation of spatial field of interest

The first main point of this paper is to incorporate the field estimation technique into the task space control of manipulators to achieve target localization with the end-effector. The desired motion command is resolved on-line through the gradient information learned up to the present time. Specifically, we will adopt the approach using the RBF network as a mathematical model of regional field.[5]

Consider a differentiable spatial basis function vector $h(\xi) : \mathbb{R}^\ell \to \mathbb{R}^m$, where $\xi \in \mathbb{R}^\ell$ is the coordinate vector in ℓ-dimensional space and m denotes the total number of basis functions used to approximate the field with reasonable accuracy within the surveillance region. Then we can represent the true field of interest denoted by $\mathcal{T}(\xi) : \mathbb{R}^\ell \to \mathbb{R}$ as a weighted sum of basis functions as

$$\mathcal{T}(\xi) := \theta^T h(\xi) = \sum_{i=1}^{m} \theta_i h_i(\xi), \tag{13}$$

where $\theta := [\theta_1 \quad \theta_2 \quad \cdots \quad \theta_m]^T$ is a weight vector, which parameterizes the field. $h_i(\xi) = \bar{h}_i(\|\xi - \mu_i\|)$ is the ith RBF centered at a location $\mu_i \in \mathbb{R}^\ell$ in the surveillance region. We assume the linear form of the basis function network, which is intended to utilize the standard RLS technique as our learning tool. Any set of (differentiable) functions, the response of which changes monotonically with distance from the center point can be used as basis functions. In describing environmental or biological behaviors, a typical choice is the Gaussian RBF, which can be written as[5]

$$h_i(\xi) = \exp\left(-\frac{\|\xi - \mu_i\|^2}{\sigma^2}\right), \tag{14}$$

where σ denotes the common width of basis functions.

A set of field data is assumed to be measured by on-board sensors mounted to the manipulator links. Denoting the number of sensors by m_s, the measurement equation for the jth $\in \{1, 2, \ldots, m_s\}$ field sensor can be written as

$$\lambda_j(k) = \mathcal{T}(\xi_j^s(k)) + w_j(k), \tag{15}$$

where k is the discrete time index, i.e., $t = kT_f$ with T_f denoting the time interval to update the RLS routine, $\xi_j^s(k)$ is the coordinate of the location of the jth sensor at time k, and $w_j(k)$ is the corresponding measurement noise, which is assumed to be white and zero mean. The sensor signals and their noises are denoted in vector forms as

$$\lambda(k) := \left[\lambda_1(k) \cdots \lambda_{m_s}(k)\right]^T,$$
$$w(k) := \left[w_1(k) \cdots w_{m_s}(k)\right]^T.$$

The main goal of field estimation is to get the best estimate, $\hat{\theta}(n)$ at the present time $t = nT_f$ based on all the prior (noisy) observations $\lambda(1), \ldots, \lambda(n)$. Specifically, $\hat{\theta}(n)$ is obtained as a solution to minimize the least square error:

$$\hat{\theta}(n) = \min_{\theta \in \Theta} \sum_{k=1}^{n} \|\lambda(k) - H(k)^T\theta\|^2, \tag{16}$$

where $\Theta \subset \mathbb{R}^m$ is a compact set for parameter vector θ and $H(k) \in \mathbb{R}^{m \times m_s}$ is the basis function matrix evaluated at each sensor location, i.e,

$$H(k) := \left[h\left(\xi_1^s(k)\right) \cdots h\left(\xi_{m_s}^s(k)\right)\right]. \tag{17}$$

https://doi.org/10.1017/S0263574711001019 Published online by Cambridge University Press

The solution to Eq. (16) can be obtained in a recursive way as[18]

$$\hat{\theta}(k) = \hat{\theta}(k-1) + K(k)\left(\lambda(k) - H^T(k)\hat{\theta}(k-1)\right), \quad (18a)$$

$$K(k) = P(k)H(k) = P(k-1)H(k)\left(I_{m_s} + H^T(k) P(k-1)H(k)\right)^{-1}, \quad (18b)$$

$$P(k) = P(k-1) - P(k-1)H(k)\left(I_{m_s} + H^T(k) P(k-1)H(k)\right)^{-1} H^T(k)P(k-1) = \left(I_m - K(k)H^T(k)\right) P(k-1), \quad (18c)$$

where $I_{(\bullet)}$ is the identity matrix with size $\bullet$ and

$$P^{-1}(k) := \mathcal{H}(k)\mathcal{H}^T(k) = \sum_{i=1}^{k} H(i)H^T(i). \quad (19)$$

Note that $P(k)$ is singular if $k \times m_s < m$. So the initial condition $P_0 := P(0)$ needs to be selected as a positive definite matrix. Then

$$P(k) = \left(P_0^{-1} + \mathcal{H}(k)\mathcal{H}^T(k)\right)^{-1}, \quad (20)$$

and $P(k) \succ 0$ (i.e., positive definite) for all k. Moreover, note that $P(k)$ can be made arbitrarily close to $(\mathcal{H}^T(k)\mathcal{H}(k))^{-1}$ by choosing P_0 sufficiently large. In practice, P_0 may be chosen as large as allowed by the initial learning effort and the noise sensitivity. Typically, P_0 can be chosen as a diagonal matrix.

3.2. Field tracking with gradient climbing

Given the estimated weight vector $\hat{\theta}(k)$ by Eq. (18a), the field tracking control law Γ_f in Eq. (8) can be implemented by gradient

$$\Gamma(k) = K_f \widehat{\nabla \mathcal{T}}(x(k)) = K_f \left.\frac{\partial h(\xi)}{\partial \xi}\right|_{\xi=x(k)}^{T} \hat{\theta}(k), \quad (21)$$

where $K_f \in \mathbb{R}^{\ell \times \ell}$ is a diagonal matrix that represents a step size for the field tracking. K_f needs to be properly selected considering the intensity of field measurements and the magnitude of control force of the actuator. Note that our field model $\mathcal{T}(\xi)$ shown in Eq. (13) can be considered as a class of artificial potential functions parameterized by the unknown weight vector $\hat{\theta}(k)$ and known basis function $h(\xi)$. Thus, any additional control effort, such as the obstacle avoidance, may be augmented into the field model with required potential functions added to $\mathcal{T}(\xi)$. In practice, additional potential functions will need to be carefully scaled not to affect the above gradient force, i.e., the major force component to drive the end-effector toward the target point of the unknown field.

Equation (21) may be regarded as the desired velocity in task space. We do not have any prior knowledge of $\mathcal{T}(\xi)$, thus the gradient vector used in Eq. (21) may encounter a large control effort if there is a sudden change in the scalar field. Also, it may create undesirable vibration due to the structural elasticity of the manipulator. To avoid such a situation, the maximum limit can be imposed on the allowable desired task space velocity using the saturation function:

$$a_x(t) = K_v \bar{V} \mathbf{sat}\left(\bar{V}^{-1} K_v^{-1} K_f \left.\frac{\partial h(\xi)}{\partial \xi}\right|_{\xi=x(k)}^{T} \hat{\theta}(k)\right) - K_v \hat{v}(t), \quad (22)$$

where $\bar{V} \in \mathbb{R}^{\ell \times \ell}$ is a diagonal matrix with its elements being the magnitude of the maximum allowable task space velocities for each coordinate axis. The saturation function in Eq. (22) constrains the magnitude of the gradient force command to be less than $\bar{V}$. The saturation level needs to be selected so that the maximum speed of the end-effector does not exceed the value with which the manipulator can operate safely under the operating environment. Note that Eq. (22) uses notations for both the continuous time and the discrete time domain. As mentioned above, the discrete time index is used to emphasize the fact that the RLS routine is updated in a much slower rate than joint control signals.

3.3. Singularity tolerant task space control

This subsection presents the second main point of this paper. For the case of a redundant manipulator, Eqs. (3) and (9) lead to the joint control vector $\tau(k)$ rewritten as

$$\tau(t) = \hat{M}(t)\left\{\hat{J}^{\dagger}(t)\left(a_x(t) - \dot{\hat{J}}(t)\dot{q}(t)\right) + \left(I - \hat{J}^{\dagger}(t)\hat{J}(t)\right)\eta(t)\right\} + \hat{L}(t)\dot{q}(t) + \hat{G}(t) + \hat{F}_f(t) - \hat{F}_d(t), \quad (23)$$

with $a_x(k)$ and $\eta(k)$ given by Eqs. (22) and (11), respectively. Then the resulting closed loop system becomes[17]

$$\dot{v} = -K_v \hat{v}(t) + K_v \bar{V} \mathbf{sat}\left(\bar{V}^{-1} K_v^{-1} K_f \nabla h(x(k))\hat{\theta}(k)\right) + \mathcal{F}_p. \quad (24)$$

When a_x drives the end-effector by the estimated gradient, the manipulator may be guided toward its singular configuration. Since we are using the Cartesian coordinates as the task space variables, the singularity can be one of the two cases:

(i) *Interior singularity*: The singularities occurring when the end-effector is strictly inside of its operational range in the task space.
(ii) *Boundary singularity*: the singularities coinciding with workspace boundary.

While the interior singularity can always be avoided by the null space projection used in Eq. (9), the boundary singularity cannot be avoided if the task space command a_x keeps driving the end-effector toward the outside of the operation range. In such cases, the end-effector is not able to locate the exact target source, but it can still be positioned to point the direction where the target source, at least, gives us the correct direction toward which the moving platform has to move. The Jacobian pseudo inverse approaches infinity as it gets near the singularity. DLS introduces the directional error,[11] thus diminishing the orthogonality of the projection term in Eq. (9). The adjoint Jacobian method[12] maintains the orthogonality but suppresses the motion command along the *non-degenerate direction* (feasible direction) as well as the *degenerate direction* (infeasible direction).

https://doi.org/10.1017/S0263574711001019 Published online by Cambridge University Press

The main idea of the approach taken in this paper is based on the on-line decomposition of the command vector a_x into the *non-degenerate direction* and its *orthogonal* as the manipulator approaches the boundary singularity. The approach taken by Chang and Khatib[20] may be considered as a similar concept, where the manipulator is regarded as redundant near the singular configuration and the subspace decomposition is conducted to drive the end-effector through singularities. For the boundary singularity considered in this paper, the manipulator essentially loses its degree of freedom rather than becoming redundant. Also, due to the gradient force pulling the end-effector out of its workspace, the manipulator is expected to stay in the singular configuration onwards. So the approach taken here is to capture only the component of the taskspace force that is orthogonal to the singular direction while gradually discarding that of the singular direction as the manipulator gets closer to the singular configuration.

Let J_c denotes the collapsed Jacobian, which is formed by collecting the linearly independent columns of J. Then a_x can be rewritten as

$$a_x = a_c + a_r, \tag{25}$$

where a_c is the projection of a_x onto the range space of J_c and a_r is its orthogonal residual. Then we can redefine a_q in Eq. (9) as follows:

$$a_q = \hat{J}_\varepsilon^\dagger\left(a_x + \hat{\dot{J}}\dot{q}\right) + (1-\alpha)\begin{bmatrix}\hat{J}_c^\dagger \\ 0\end{bmatrix}\left(a_c + \hat{\dot{J}}\dot{q}\right) + \left(\alpha I - \hat{J}_\varepsilon^\dagger \hat{J}\right)\eta, \tag{26}$$

where

$$\alpha = \mathbf{sat}\left(\frac{\det(\hat{J}\hat{J}^T)}{\varepsilon}\right), \tag{27a}$$

$$\hat{J}_\varepsilon^\dagger = \hat{J}^T \sigma\left(\det(\hat{J}\hat{J}^T)\right)\mathrm{adj}(\hat{J}\hat{J}^T), \tag{27b}$$

$$\text{and } \sigma\left(\det(\hat{J}\hat{J}^T)\right) = \begin{cases} \frac{1}{\det(\hat{J}\hat{J}^T)} & \text{if } \det(\hat{J}\hat{J}^T) \geq \varepsilon \\ \frac{1}{\varepsilon} & \text{otherwise} \end{cases} \tag{27c}$$

ε denotes the threshold value that determines whether the manipulator is close to singularity or not. For simplicity, the argument of the Jacobian function (q) is omitted in the above equations. Equation (26) is equivalent to Eq. (9) if $\det(\hat{J}\hat{J}^T) \geq \varepsilon$. If $\det(\hat{J}\hat{J}^T)$ gets smaller than ε, then the first term in Eq. (26) coincides with the adjoint method, which will diminish a_x as $\det(\hat{J}\hat{J}^T)$ decreases below ε proportionally to α. However, the second term of Eq. (26) preserves the projection component of a_x on the range space of J_c. Figure 2 illustrates this point using the three-link manipulator.

In fact, J_c can be easily obtained by selecting the first few columns of J, which are linearly independent. In planar manipulation, only the first column can be obtained, whereas in the three dimensional manipulation, the first two columns will do the work if the first two joints are orthogonal revolute joints as in typical industrial manipulators.

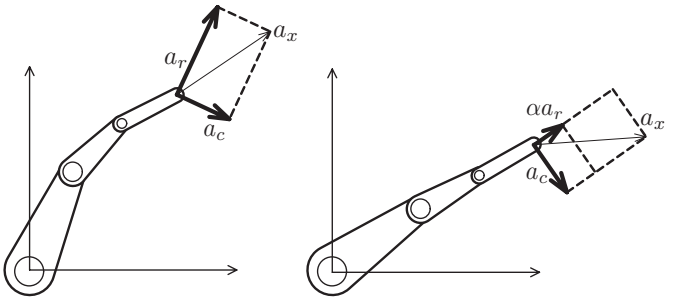


Fig. 2. Task space command adjustment by singularity tolerant scheme. (a) $\det(\hat{J}\hat{J}^T) \geq \varepsilon$; (b) $\det(\hat{J}\hat{J}^T) < \varepsilon$.

3.4. Convergence analysis

This subsection presents a brief convergence analysis for the task space control law combined with the recursive field estimation scheme. For the simplicity of analysis, we consider only the ideal case without parameter uncertainties (Eq. (7)) and the gradient saturation (Eq. (22)).

We start analysis by deriving an alternate expression of $\tilde{\theta}(k)$. From Eq. (18) we can see that

$$\begin{aligned}\tilde{\theta}(k) &= \tilde{\theta}(k-1) + P(k)H(k)\left(H^T(k)\theta + w(k) - H^T(k)\,\hat{\theta}(k-1)\right)\\ &= \tilde{\theta}(k-1) + P(k)H(k)\left(w(k) - H(k)^T\tilde{\theta}(k-1)\right)\\ &= \left(I - P(k)H(k)H^T(k)\right)\tilde{\theta}(k-1) + P(k)H(k)w(k)\\ &= P(k)\left(P^{-1}(k) - H(k)H^T(k)\right)\tilde{\theta}(k-1) + P(k)H(k)w(k)\\ &= P(k)P^{-1}(k-1)\tilde{\theta}(k-1) + P(k)H(k)w(k),\end{aligned} \tag{28}$$

where the last equality is from Eq. (19). Accordingly,

$$\begin{aligned}P^{-1}(k)\tilde{\theta}(k) &= P^{-1}(k-1)\tilde{\theta}(k-1) + H(k)w(k)\\ &= P^{-1}(0)\tilde{\theta}(0) + \sum_{j=1}^{k} H(j)w(j),\end{aligned} \tag{29}$$

where the second equality is obtained by induction. Consequently, from Eq. (20) we arrive at

$$\tilde{\theta}(k) = \left(P_0^{-1} + \mathcal{H}(k)\mathcal{H}^T(k)\right)^{-1}\left(P_0^{-1}\tilde{\theta}(0) + \sum_{j=1}^{k} H(j)w(j)\right). \tag{30}$$

Then the mean and variance of the estimation error become

$$E\left(\tilde{\theta}(k)\right) = \left(P_0^{-1} + \mathcal{H}(k)\mathcal{H}^T(k)\right)^{-1} P_0^{-1}\tilde{\theta}(0), \tag{31a}$$

$$\mathrm{var}\left(\tilde{\theta}(k)\right) = \left(P_0^{-1} + \mathcal{H}(k)\mathcal{H}^T(k)\right)^{-1}\mathcal{H}(k)W\mathcal{H}^T(k) \times \left(P_0^{-1} + \mathcal{H}(k)\mathcal{H}^T(k)\right)^{-1}, \tag{31b}$$

https://doi.org/10.1017/S0263574711001019 Published online by Cambridge University Press

where W is the covariance of $w_j(k)$. If we assume that the persistence excitation (PE) condition is met, then

$$\exists\, N \text{ and } \epsilon > 0 \quad \text{such that} \quad \sum_{i=k}^{k+N-1} H(i)H^T(i) \succeq \epsilon I_m \quad \forall k, \tag{32}$$

where $\succeq$ denotes the matrix inequality in the sense of positive semi-definiteness, and we can see from Eq. (31) that $\tilde{\theta}(k)$ asymptotically converges to zero with probability one. Furthermore, under a uniform PE condition (i.e., $H(k)H(k)^T$ is almost constant), $\tilde{\theta}(k)$ will converge at the rate $1/(km_s)$, which shows that the convergence rate is proportional to the number of sensors.

Now let us consider the following continuous time Lyapunov function, which is the summation of the kinetic energy of the manipulator and the negative potential of the field:

$$V(t) = \frac{v(t)^T K_f^{-1} v(t)}{2} + (\mathcal{T}_{\max} - \mathcal{T}(x(t))), \tag{33}$$

where $\mathcal{T}_{\max}$ is the maximum value of $\mathcal{T}(x(t))$. It is easy to see that $V(t) \geq 0$, since $K_f \succ 0$ and $\mathcal{T}_{\max} \geq \mathcal{T}(x(t))$. Assuming no perturbation (i.e., $\mathcal{F}_p = 0$, $\hat{x} = x$, and $\hat{v} = v$) and also disregarding the saturation, the closed loop dynamics in Eq. (24) becomes $\dot{v}(t) = -K_v v(t) + K_f \nabla h(x(k))\hat{\theta}(k)$. In such a case, the time derivative of the Lyapunov function can be written as

$$\frac{dV(t)}{dt} = -v^T(t)K_f^{-1}K_v v(t) + v^T(t)\nabla h^T(x(t))\tilde{\theta}(k), \tag{34}$$

which is negative outside the ball

$$\|v(t)\| \leq \frac{k_v^{\max}}{k_f^{\min}} \|\nabla h^T(x(k))\| \|\tilde{\theta}(k)\|, \tag{35}$$

where $k_v^{\max}$ and $k_f^{\min}$ denote the maximum and the minimum of the diagonal elements of K_v and K_f, respectively. Note that this ball can be found from Eq. (34) by the region that will make $\frac{dV(t)}{dt} \leq 0$. Since $\tilde{\theta}(k)$ converges to 0 with probability one (Eq. (31)), $v(t)$ will ultimately be bounded by 0 with probability one. From Eqs. (5) and (8), $v(t) \equiv 0$ implies $\nabla h^T(x(k))\theta \equiv 0$. Therefore,

$$x(t) \to \xi^* \quad \text{as } t \to \infty, \tag{36}$$

where ξ^* denotes the unique maximum point of $\mathcal{T}(\xi)$. Certainly, the assumption on the PE condition (Eq. (32)) plays a critical role in assuring the convergence.

3.5. Application issues

The main part of the control scheme proposed in this paper can be characterized as the spatial field estimation using the *deterministic* (or *parametric*) field model. It should be reminded that the RBF model adopted in this paper is one example of deterministic field model. Such a scalar field can come in many different forms in real life applications (e.g., chemical diffusion, radioactive propagation, and magnetic or optical fields) and prior knowledge on the nature of the target field may favor different basis functions. Therefore, the effectiveness of the field estimation scheme will rely on the selection of proper field model and its parameters (i.e., the number and the geometric properties of the basis functions). In addition to using the proposed scheme for the autonomous manipulation with the unknown source tracking, it may also be used for sensor-guided robotic manipulation where the physical field (and its source) is created by the user for the purpose of guiding the end-effector to a location randomly specified by the user action. In such applications, the user will be equipped with the full knowledge on the correct field model. In either case, there exists a trade-off between the correctness of the field model and the computational complexity.

As we can see, the performance of target localization and tracking of the robot manipulator is limited by its operation range. Running several manipulators (potentially with mobility) in a coordinated fashion will greatly extend the effectiveness of the proposed control scheme.

Table I. Parameters for the simulation.

Plant parameters	Values (1st, 2nd, 3rd)	Units
Link length	0.36, 0.24, 0.2 (m)	
Motor torque const.	24.3, 4.89, 1.5	(Nm/V)
Viscous fric.	0.03, 0.02, 0.015	(Nms/rad)
Stiction	0.42, 0.4, 0.3	(V)
Mass	9.214, 4.507, 3.5	(kg)
Inertia, J	0.224, 0.05, 0.04	(kgm^2)
Number of sensors	4(2)	
Field model para.	**Value**	**Units**
Number of bases, m	3×3	
Sensor noise, W	0.1	
Width of RBF, σ	0.35	
Surveillance region, x	$[-0.75, 0.75]$	(m)
surveillance region, y	$[-0.25, 0.75]$	(m)
Control para.	**Value**	**Units**
Maximum velocity, $\bar{v}$	diag(0.5, 0.5)	(m/sec)
Damping factor, K_v	diag(44, 44)	(Ns/m· kg)
Field gain, K_f	diag(5, 5)	(N)
$P(0)$	$10 \cdot I_{9\times 9}$	
$\theta(0)$	$0_{9\times 1}$	
ε	0.01	

4. Simulation

The proposed control method is applied to a three-link planar manipulator deployed to a two-dimensional spatial region to search for a source location of the field. The first joint of the robot is assumed to be positioned at the origin of the spatial coordinate frame. Parameters for the simulation are summarized in Table I. The decoupling feedback control law is run at the base sample rate of $T_s = 1$ ms, while the field estimation is run at $T_f = 5$ ms.

Figure 3 shows the end-effector trajectory with the initial position at $x(0) = (-0.4, 0)$ when the redundancy is not

https://doi.org/10.1017/S0263574711001019 Published online by Cambridge University Press

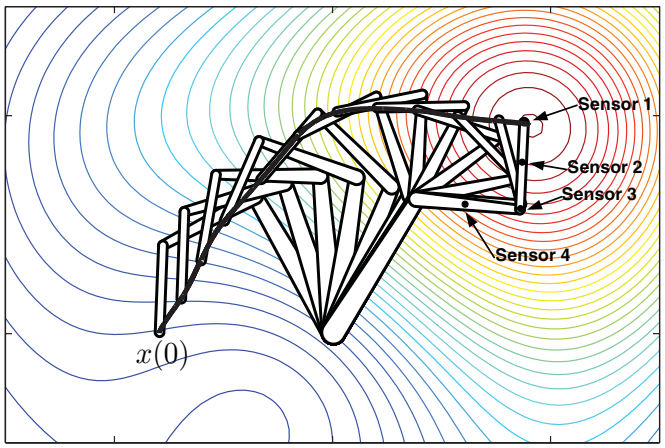


Fig. 3. (Colour online) End-effector trajectory for $x(0) = (-0.4,\ 0)$ ($\eta = 0$).

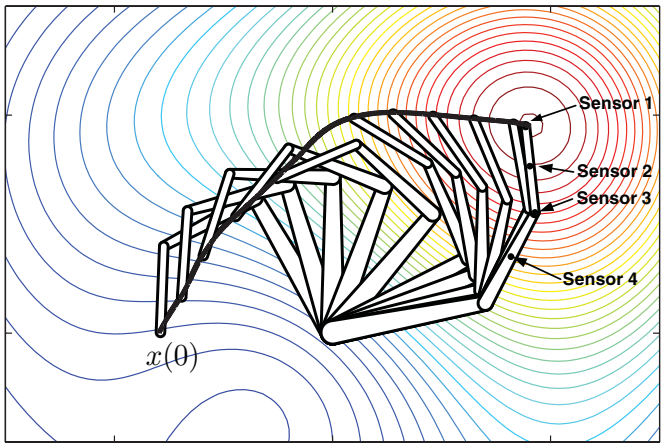


Fig. 4. (Colour online) End-effector trajectory for $x(0) = (-0.4,\ 0)$ with manipulability constraint.

accounted for, i.e., $\eta = 0$. Note that the three-link serial chain is considered here as a redundant manipulator because only the Cartesian coordinates $(x,\ y)$ are used as task space variables. Four sensors are mounted at the locations shown in the final configuration of the manipulator. More specifically, the locations of sensors are the center of the second link, the third joint, the center of the third link, and the end tip of the third link. The spatial distribution of the unknown field is represented by the contour lines. It is observed that the end-effector is tracking the gradient direction of the field quite well from the beginning, which suggests that the convergence of the parameter estimation is achieved at earlier stage of the navigation. As the end-effector approaches the target source, the first joint slows down and the second joint provides major movement to bring the end-effector to the target source.

Figure 4 shows the manipulation profiles using the same field model and initial position but with the manipulability constraint imposed on the potential function given in Eq. (12). Compared with Fig. 3, the first joint is mainly responsible for the end-effector motion along the sequence of motions as it tries to keep the joint angles away from the singular configuration. For the trajectory given in Fig. 4, the estimation errors are plotted in Fig. 5. The upper plot is the norm of the parameter estimation error and the lower plot shows the estimation error of the gradient. We can see that

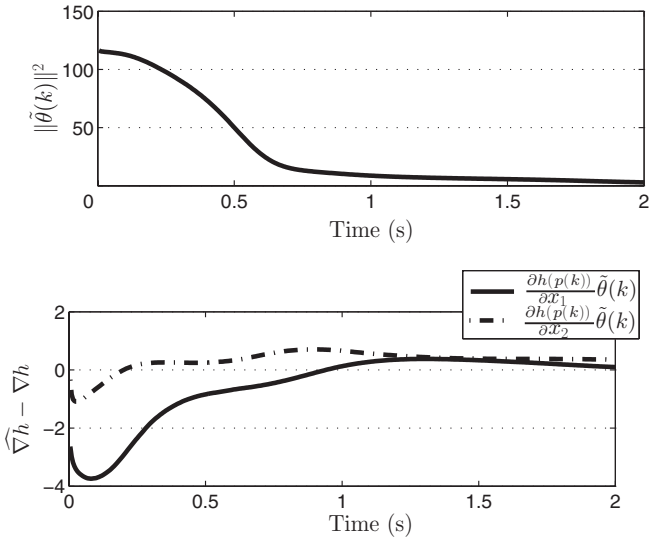


Fig. 5. Convergence of estimation errors for the results of Fig. 4. (a) Parameter estimation error; (b) gradient estimation error.

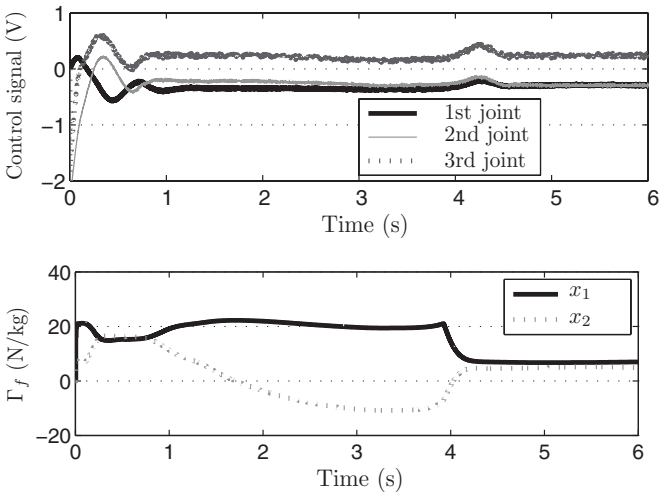


Fig. 6. Control forces for the results of Fig. 4. (a) Joint control signals; (b) gradient force for field tracking (Γ_f in Eq. (8)).

the convergence of parameter estimation error $\tilde{\theta}(k)$ mostly occurs within 1 s and it keeps decreasing at much slower rate after 1 s.

The actual control torques (τ) and the gradient forces (Γ_f) for the results in Fig. 4 are plotted in Fig. 6. The movement of the end-effector reaches its steady-state a little after 4 s. Both control forces and Γ_f possess non-zero steady-state values because of the static (Coulomb) friction forces at each joint.

In order to see the effect of number of sensors, the simulation results with two sensors are shown in Fig. 7. One sensor is mounted at the end-effector and the other is at the third joint. As can be seen from Fig. 7, the overall trajectory of the end-effector does not differ much from the case with four sensors. However, Fig. 8 reveals that the convergence of the parameter error has been delayed by about half second. The magnitudes of the gradient errors have also increased as shown in Fig. 8(b).

In order to see the effect of the initial position, the end-effector is started at $x(0) = (0,\ -0.2)$ and its trajectory and parameter convergence plots are given in Figs. 9 and

https://doi.org/10.1017/S0263574711001019 Published online by Cambridge University Press

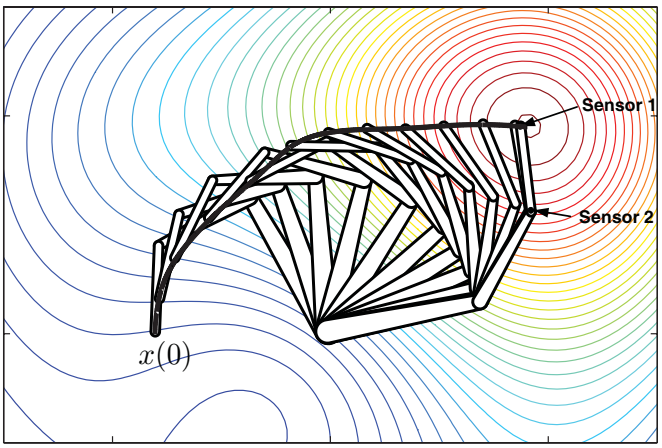


Fig. 7. (Colour online) End-effector trajectory using two sensors.

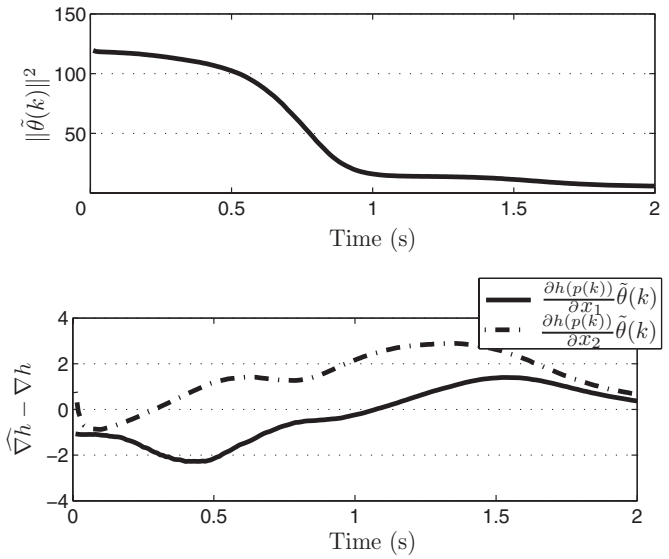


Fig. 8. Convergence of estimation errors for the results of Fig. 7. (a) Parameter estimation error; (b) gradient estimation error.

10, respectively. The major parameter convergence took place between 1 s and 1.5 s, which shows that the speed of convergence is further increased compared with Fig. 7. For this particular case, the *a priori* output prediction error $\tilde{\lambda}(k) = \lambda(k) - h^T(k)\hat{\theta}(k)$ is plotted separately in Fig. 11. After its initial drop at the first recursive step ($k = 1$), the parameter error remains very small until around 0.5 s, which explains slow convergence in the initial period. As can be seen in Eq. (18a), the *a priori* output prediction error determines the amount of correction on the estimated parameter. Thus, the small $\tilde{\lambda}(k)$ in the initial period accounts for the slow convergence as shown in Fig. 10(a). Likewise, the increase in $\tilde{\lambda}(k)$ around 1 s explains the expedited parameter convergence after 1 s shown in Fig. 10(a).

The effectiveness of the singularity tolerant scheme has been illustrated in Fig. 12. $\varepsilon = 0.01$ is chosen as the threshold value to determine the singularity. Since the target source is intentionally located outside the workspace, the manipulator is driven close to its boundary singularity (fully stretched configuration) during the final stage of the motion. Despite the singularity, the manipulator is still controlled to move along the non-degenerate direction through the movement of

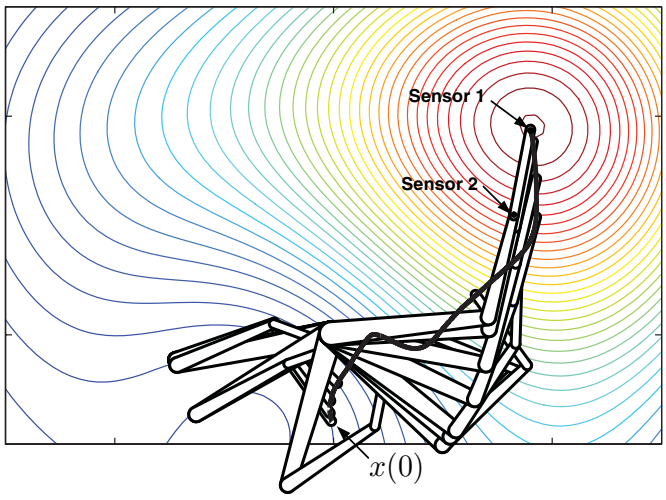


Fig. 9. (Colour online) End-effector trajectory for $x(0) = (0, -0.2)$.

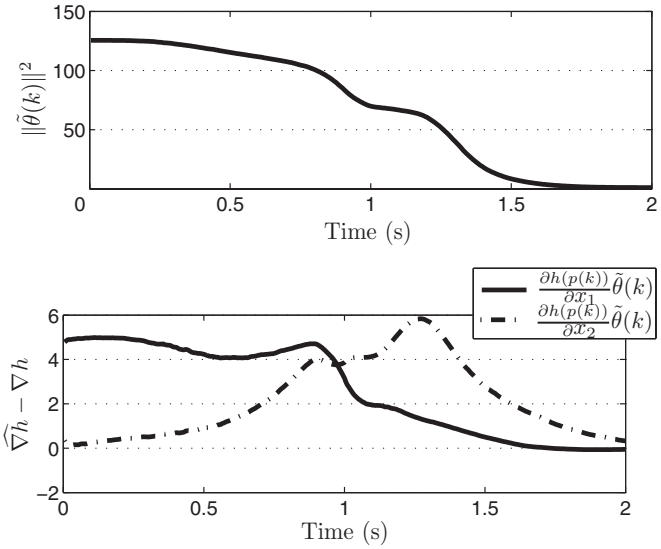


Fig. 10. Convergence of estimation errors for the results of Fig. 9. (a) Parameter estimation error; (b) gradient estimation error.

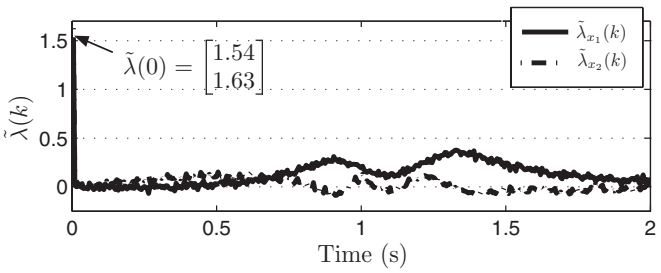


Fig. 11. The *a priori* output prediction error for the results of Fig. 9.

the first joint. Eventually, the manipulator is configured to point the direction along which the target source is located.

Finally, to demonstrate the proposed method in the general three-dimensional space, the simulation results with 6R elbow manipulator is presented in Figs. 13 and 14. The elbow manipulator started from the initial position at $x(0) = [0, -0.7, 0]$ and driven toward the target point near $\xi^* = [-0.2, 0.2, 0.45]$. Figure 13 shows the initial posture

https://doi.org/10.1017/S0263574711001019 Published online by Cambridge University Press

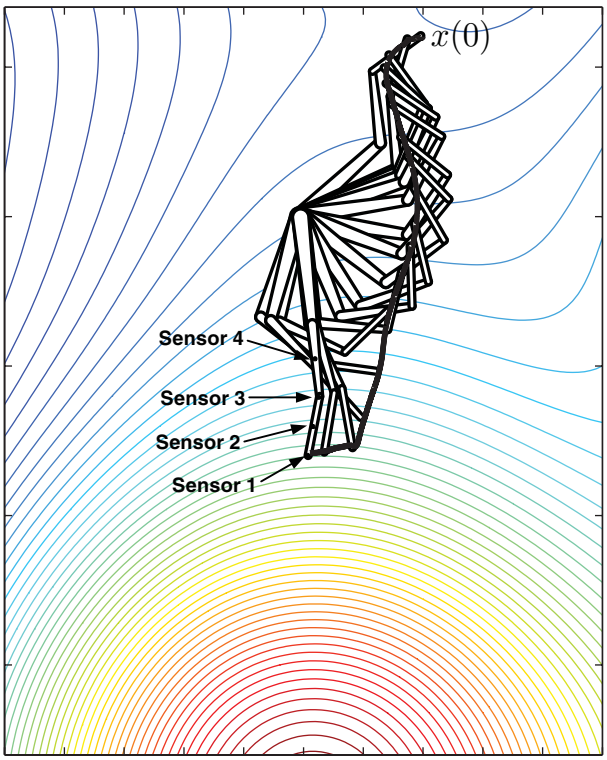


Fig. 12. (Colour online) Singularity tolerant end-effector trajectory for $x(0) = (0.4,\ 0.6)$.

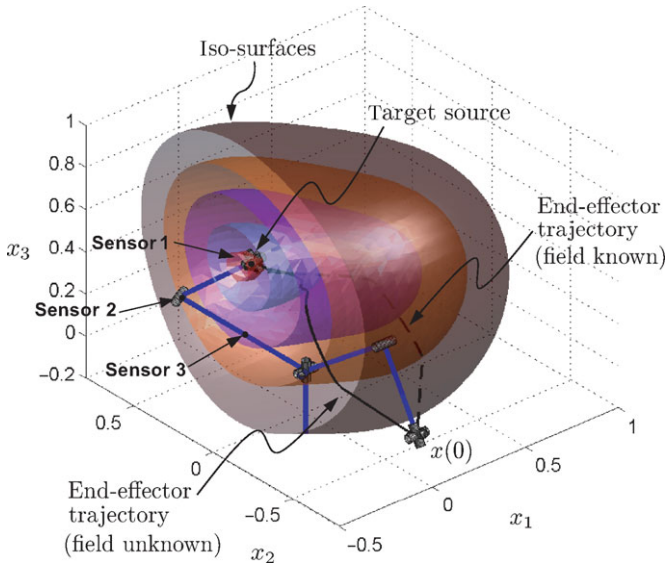


Fig. 13. (Colour online) End-effector trajectory for elbow manipulator with $x(0) = (0,\ -0.7,\ 0)$.

at $t = 0$ s and the final posture at $t = 8$ s, as well as the trace of the end-effector during this period. Cross-sectional view of five different levels of the iso-surfaces are drawn to show how the environmental potential field is structured. The potential field is modeled with 8 ($2 \times 2 \times 2$) basis functions. Note that the three joints at the wrist are regarded as redundant degrees of freedom, since the target source is characterized by the Cartesian coordinates only.

In Fig. 13, the dashed trajectory line represents the end-effector motion when the potential function of the field is assumed to be fully known. In other words, the dashed

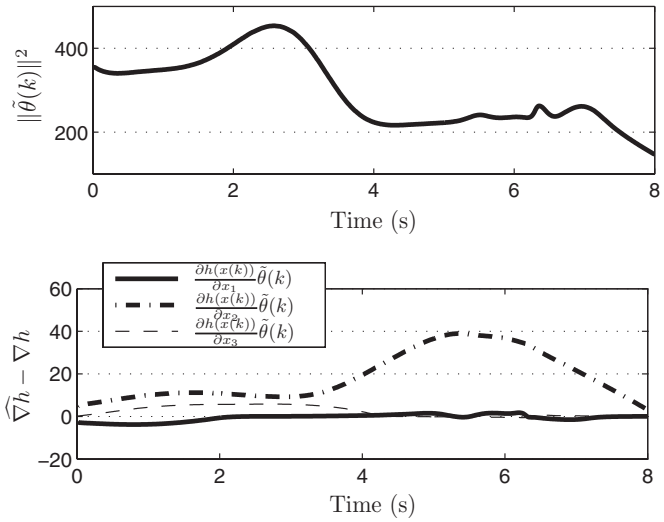


Fig. 14. Convergence of estimation errors for the results of Fig. 13. (a) Parameter estimation error; (b) gradient estimation error.

line indicates the trajectory when we apply the conventional potential function approach. Comparing the solid line trajectory with the dashed one, we can see that the lack of knowledge on the field model makes the trajectory initially deviated from the ideal path. However, as the scalar sensors collect more data, the gradient vector in Eq. (21) converges to the true gradient. As a result, the solid line trajectory near the target source coincides with the dashed line.

The animation movie for the results in Fig. 13 is available at the following website: http://mme.uwaterloo.ca/%7Esoojeon/movie.html

5. Conclusions and Future Work

This paper presents a new task space control design of a robot end-effector to achieve the source tracking in an unknown field. Using the RBF network model, the unknown field is estimated by the RLS method and the estimated field information is used for the end-effector to navigate through the unknown field and to track the gradient direction. To handle the situation where the manipulator is driven toward its singular configuration, a singularity tolerant task space control scheme is proposed so that the manipulator can still be controlled along the non-degenerate direction. Convergence analysis using the Lyapunov stability theory showed that the end-effector will asymptotically converge to the target source under the PE condition, which is also demonstrated through simulation results. The number of sensors is shown to be proportional to the rate of convergence of the field parameters. In practice, the recursive estimation needs to be implemented in much slower time scale than the joint control. The results in this paper will be useful in extending the application of articulated manipulation to various autonomous operation tasks.

Various experimental tests are underway to validate the real-life performance of the proposed control method. One of preliminary tests relevant to this paper can be found in the recent report[23] where a smooth gray scale map is adopted to

https://doi.org/10.1017/S0263574711001019 Published online by Cambridge University Press

represent the unknown physical potential field with its scalar values measured by color sensors mounted on a two-link planar robot. Experimental tests with more realistic scenarios are currently being investigated using real scalar fields such as magnetic or light intensity.

References

1. L. E. Parker and J. V. Draper, "Robotics applications in maintenance and repair," **In**: *Handbook of Industrial Robotics*, 2nd ed. (Shimon Y. Nof, ed.) (John Wiley, New York, 1998).
2. J. Vertut and P. Coiffet, *Teleoperation and Robotics: Applications and Technology* (Prentice-Hall. Englewood Cliffs, NJ, 1985).
3. K. Erik, L. Pål and T. A. Andreas, "A Robotic Concept for Remote Inspection and Maintenance on Oil Platforms," **In**: *Proceedings of the ASME 28th International Conference on Ocean, Offshore and Arctic Engineering*, Honolulu, Hawaii, USA (2009) pp. 667–674.
4. K. Glass, R. Colbaugh and H. Seraji, "Real-Time Control for a Serpentine Manipulator," **In**: *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, Grenoble, France (1997) pp. 1775–1780.
5. J. Choi, S. Oh and R. Horowitz,"Distributed learning and cooperative control for multi-agent systems," *Automatica* **45**(12), 2802–2814 (2009).
6. S. Martinez, "Distributed interpolation schemes for field estimation by mobile sensor networks," *IEEE Trans. Control Syst. Technol.* **18**(2), 491–500 (2009).
7. N. Leonard, D. A. Paley, F. Lekien, R. Sepulchre, D. M. Fratantoni and R. E. Davis, "Collective motion, sensor networks, and ocean sampling," *Proc. IEEE* **95**(1), 48–74 (2007).
8. D. Grünbaum, "Schooling as a strategy for taxis in a noisy environment," *Evolution. Ecol.* **12**, 503–522 (1998).
9. V. N. Christopoulos and S. I. Roumeliotis, "Adaptive Sensing for Instantaneous Gas Release Parameter Estimation," **In**: *Proceedings of the IEEE International Conference on Robotics and Aotumation*, Barcelona, Spain (2005) pp. 4450–4456.
10. O. Khatib, "Real-time obstacle avoidance for manipulators and mobile robots," *Int. J. Robot. Res.* **5**(1), 90–98 (1986).
11. Y. Nakamura and H. Hanafusa, "Inverse kinematics solutions with singularity robustness for robot manipulator control," *ASME J. Dyn. Syst. Meas. Control* **108**, 163–171 (1986).
12. Y. Tsumaki and D. N. Nenchev, "Jacobian adjoint matrix-based approach to teleoperation," **In**: *Proceedings of the International Symposium on Microsystems, Intelligent Materials and Robots*, Sendai, Japan (1995) pp. 532–535.
13. D. Oetomo and M. H. Ang Jr., "Singularity robust algorithm in serial manipulators," *Robot. Comput.-Integr. Manuf.* **25**, 122–134 (2009).
14. O. Khatib, "Inertial Properties in robotic manipulation: An object-level framework," *Int. J. Robot. Res.* **14**(1), 19–36 (1994).
15. T. Yoshikawa, "Analysis and control of robot manipulators with redundancy," **In**: *Robotics Research: 1st International Symposium* (M. Brady and R. Paul, eds.) (MIT Press, Cambridge, MA, 1984) pp. 735–747.
16. L. Sciavicco and B. Sicilino, *Modeling and Control of Robot Manipulators*, 2nd ed. (McGraw-Hill, New york, NY, 2000).
17. M. W. Spong, S. Hutchinson and M. Vidyasagar, *Robot Modeling and Control* (John Wiley, New York, 2006).
18. K. J. Åström and B. Wittenmark, *Adaptive Control*, 2nd ed. (Addison-Wesley, Boston, MA, 1995).
19. R. V. Patel and F. Shadpey, *Control of Redundant Robot Manipulators: Theory and Experiments* (Springer-Verlag, Germany, 2005).
20. K. S. Chang and O. Khatib, "Manipulator Control at Kinematic Singularities: A Dynamically Consistent Strategy," **In**: *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems*, Pittsburgh, Pennsylvania, USA (August 5–9, 1995) pp. 84–88.
21. J. Nakanishi, R. Cory, M. Mistry, J. Peters and S. Schaal, "Operational space control: A theoretical and empirical comparison," *Int. J. Robot. Res.* **27**(6), 737–757 (2008).
22. C. C. Cheah, "Task-space PD control of robot manipulators: Unified analysis and duality property," *Int. J. Robot. Res.* **27**(10), 1152–1170 (2008).
23. S. Jeon and H. J. Ahn, "Autonomous Manipulation Combining Task Space Control with Recursive Field Estimation," *Proceedings of the 2nd International Conference on Autonomous and Intelligent Systems (AIS)*, Burnaby BC, Canada (2011).

https://doi.org/10.1017/S0263574711001019 Published online by Cambridge University Press