Constrained Motion Control Using Vector Potential Fields

Samer A. Masoud and Ahmad A. Masoud

***Abstract*—This paper discusses the generation of a control signal that would instruct the actuators of a robotics manipulator to drive motion along a safe and well-behaved path to a desired target. The proposed concept of navigation control along with the tools necessary for its construction achieve this goal. The most significant tool is the artificial vector potential field which shows a better ability to steer motion than does a scalar potential field. The synthesis procedure emphasizes flexibility so that the effort needed to modify the control is commensurate with the change in the geometry of the workspace. Theoretical development along with simulation results are provided.**

***Index Terms*—Motion planning, nonlinear control, robotics manipulators, vector potential fields.**

NOMENCLATURE

NC	Navigation control.
PF	Potential field.
VPF	Vector potential field.
SPF	Scalar potential field.
BVP	Boundary value problem.
VBVP	Vector boundary value problem.
SBVP	Scalar boundary value problem.
BEM	Boundary element method.
LAC	Local alignment control.
PPC	Penetration prevention control.
BLAC	Boundary local alignment control.
BPPC	Boundary penetration prevention control.
$\mathbf{V}$	Vector potential field.
V	Scalar potential field.
$\mathbf{A}$	Vector potential field with a zero gauge ($\nabla \cdot A \equiv 0$).
$\nabla(\,)$	Gradient operator.
$\nabla X(\,)$	Curl operator.
$\nabla \cdot (\,)$	Divergence operator.
O_{old}	Pre-existing set of obstacles.
O	Newly introduced obstacles.
O_{new}	$O_{\text{old}} \cup O$.
O_{δ_d}	Region surrounding O.
O_δ	Region surrounding both O and $O_{\delta d}$.
O_v	Avoidance region in the velocity space.
ϕ	Empty set.
δd	Minimum width of the $O_{\delta d}$ region.
Γ	Boundary of O ($\Gamma = \partial O$).
Γ_δ	Boundary of O_δ.
Γ_{δ_d}	Boundary of O_{δ_d}.
Γ_v	Boundary of O_v.
Q	Image of Γ.
t	Time.
$\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}}$	Position, velocity, and acceleration vectors in the natural coordinates of the robot.
$\mathbf{x}, \dot{\mathbf{x}}$	$[\mathbf{q}^t \;\; \dot{\mathbf{q}}^t]^t$, $[\dot{\mathbf{q}}^t \;\; \ddot{\mathbf{q}}^t]^t$ respectively.
$\mathbf{e}_n(\mathbf{q})$	Unit basis vector orthogonal to Γ.
$\mathbf{e}_t(\mathbf{q})$	Unit basis vector tangent to Γ.
$\mathbf{e}_n(\dot{\mathbf{q}})$	Unit basis vector orthogonal to Γ_v.
$\mathbf{e}_t(\dot{\mathbf{q}})$	Unit basis vector tangent to Γ_v.
$\mathbf{e}_\rho$	Radial unit vector in Q.
$\mathbf{e}_\tau$	Unit vector in Q normal to $\mathbf{e}_\rho$.
$\mathbf{q}_r$	Reference vector (point) in the position space.
$q_n, \dot{q}_n$	Inner product of $\mathbf{q}$, and $\dot{\mathbf{q}}$ with $\mathbf{e}_n(\mathbf{q})$ respectively.
$\xi, \dot{\xi}$	Position vector and its time derivative in the polar coordinate system in Q which is embedded in Γ.
ξ_r	Reference point in Q.
$\mathbf{Q}(\mathbf{q})$	General phase vector field.
$\mathbf{Q}_n(\mathbf{q})$	Phase field that is normal to Γ, $\Gamma_{\delta d}$, and Γ_δ.
$\mathbf{Q}_t(\mathbf{q})$	Phase field tangent to Γ, $\Gamma_{\delta d}$, and Γ_δ and normal to $\mathbf{Q}_n$.
$M(\mathbf{q})$	Magnitude field that is modulating the strength of $\mathbf{Q}(\mathbf{q})$.
$M_n(\mathbf{q})$	Magnitude field modulating the strength of $\mathbf{Q}_n(\mathbf{q})$.
$M_t(\mathbf{q})$	Magnitude field modulating the strength of $Q_t(\mathbf{q})$.
G, Ξ	Liapunov functions.
$\dot{G}, \dot{\Xi}$	Their time derivatives.
$\mathbf{D}(q)$	Robot's inertia matrix.
$\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})$	$\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})$ Vector containing the coriolos and centripetal forces of the robot.
$\mathbf{g}(\mathbf{q})$	Gravity vector.
$\mathbf{f}(\mathbf{q}, \dot{\mathbf{q}})$	An N-D vector function.
$\mathbf{K}, \mathbf{B}$	Positive definite matrices.
ζ	Damping coefficient.
$\mathbf{u}$	Input torque vector to the robot.
$\mathbf{u}_d$	Damping component of the torque vector.
$\mathbf{u}_g$	Global component of the torque that is responsible for driving the robot to the target.
$\mathbf{u}_l$	Local component of the torque which prevents the robot from entering the newly introduced forbidden regions without blocking its motion toward the target (steering control).
$\mathbf{u}_{ln}$	Penetration prevention component of $\mathbf{u}_l$.
$\mathbf{u}_{lt}$	Local alignment component of $\mathbf{u}_l$.
$\mathbf{u}_{\Gamma l}$	Value of $\mathbf{u}_l$ at Γ.
$\mathbf{u}_{\Gamma ln}$	Value of $\mathbf{u}_{ln}$ at Γ.
$\mathbf{u}_{\Gamma l_t}$	Value of $\mathbf{u}_{lt}$ at Γ.

Manuscript received November 3, 1996; revised February 9, 2000. This paper was recommended by Associate Editor C. P. Neuman.

The authors are with the Mechanical Engineering Department, Jordan University of Science and Technology, Irbid, Jordan.

Publisher Item Identifier S 1083-4427(00)03580-3.

1083-4427/00$10.00 © 2000 IEEE

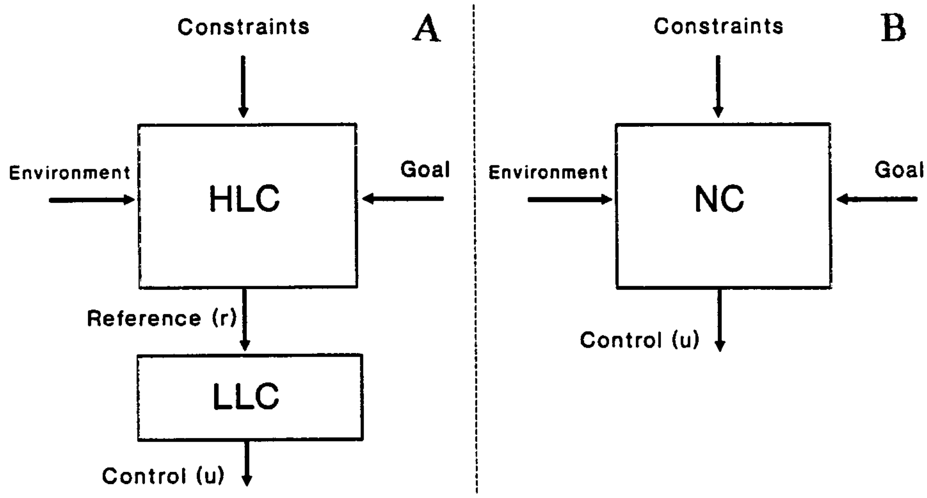


Fig. 1. (a) Classical HLC-LLC setting. (b) Navigation control.

$\alpha_i(\,)$ Scalar positive function that is modulating the strength of the normal unit vectors ($i = 1$ when used with $\mathrm{e}_n(\mathbf{q})$, $i = 2$ with $\mathrm{e}_n(\dot{\mathbf{q}})$).

$\beta_i(\,)$ Same as α_i, but used with the tangent unit vectors.

I. INTRODUCTION

WHAT makes an agent (robot) useful is its ability to exhibit a yielding purposive behavior. Yielding to the influence of an external agent (usually a human operator) may be achieved by equipping the robot with a certain class of intelligent motion controllers that are called motion planners. The setting for constructing such controllers has remained reliant on a high level controller (HLC) that utilizes classical or evolutionary AI techniques to convert the goal of the robot, the constraints on its behavior, and the information about its environment into a sequence of reference commands which are in turn fed to a classical low level controller (LLC) whose function is to generate a control signal enabling the robot to follow the reference set by the HLC [Fig. 1(a)].

One shortcoming of the aforementioned setting is the lack of guarantees that the HLC generated reference can be converted into a successful control action by the LLC. To get around this difficulty, a new fundamentally different class of controllers is needed to integrate the function of both the HLC and LLC in one control module [Fig. 1(b)]. A controller that can achieve such integration is called a navigation control (NC).

Classical controllers (LLC's) are only concerned with reference following, a behavior that is local in nature, detached from any context, and wholly dependent on the HLC for meaning and success. The behavioral difference between the NC and a classical controller may be directly observed from the arguments of their respective control functions. For a dynamical system of the form

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \mathbf{u}) \tag{1}$$

a conventional controller has the form

$$\mathbf{u} = \mathbf{h}(\mathbf{x}, \mathbf{r}, \hat{\mathbf{f}}). \tag{2}$$

On the other hand, a NC has the form

$$\mathbf{u} = \mathbf{h}(\mathbf{x}, \hat{\mathbf{f}}, \Gamma, tk) \tag{3}$$

where $\mathbf{x} \in R^{\mathbf{M}}$ is a point in the state space of the system (state vector), $\dot{\mathbf{x}}$ is its time derivative, $\mathbf{u} \in R^N$ is the control input vector of the system, $\mathbf{h} \in R^N$ is a vector function, $\mathbf{f} \in R^{\mathbf{M}}$ characterizes the dynamics of the system, $\hat{\mathbf{f}}$ is an estimate of $\mathbf{f}$, Γ is a description of the external environment of the robot, $\mathbf{r}$ is the reference to be tracked by the classical controller and $\mathbf{tk}$ describes the task which the NC is required to help the robot achieve. In a classical controller, $\mathbf{r}$ is seen as a unit in a series of local references which, if executed in the proper sequence, realize the task. In the NC, $\mathbf{r}$ does not explicitly appear in the argument of the control; rather, it implicitly generates the $\mathbf{r}s$ (in the proper sequence) from $\hat{\mathbf{f}}$, Γ, and $\mathbf{tk}$.

The tools for constructing a NC differ fundamentally from those used by classical controllers. Classical controllers use rigid, whole-domain control functions that are unequipped to comply with the stringent behavioral constraints a robot requires for successful purposive behavior. Instead, a NC generates the control action by operating on a potential field with a vector partial differential operator that functions to induce a dense set of infinitesimal actions (controls) that homogeneously cover the agent's domain of viability (workspace). This results in a freely-configurable vector being assigned to each point belonging to the workspace ($\mathbf{u}(\mathbf{x})$). A structure for the control vector group has to be determined so that the resulting solution trajectory conforms to the *a priori* specified differential and state constraints (a valid group structure.) The proper structure for the control field is what convert the infinitesimal controls into one functional unit that instructs the robot on how to reach the goal and satisfy behavioral constraints. Fig. 2(a) shows a control group structure for the simple dynamical system $[dx/dt\ \ dy/dt]^t = [ux\ \ uy]^t$ developing into a valid structure. The resulting structure is able to drive the state of the system to the target set while avoiding undesired regions in state space. Having the freedom to specify independently a control vector

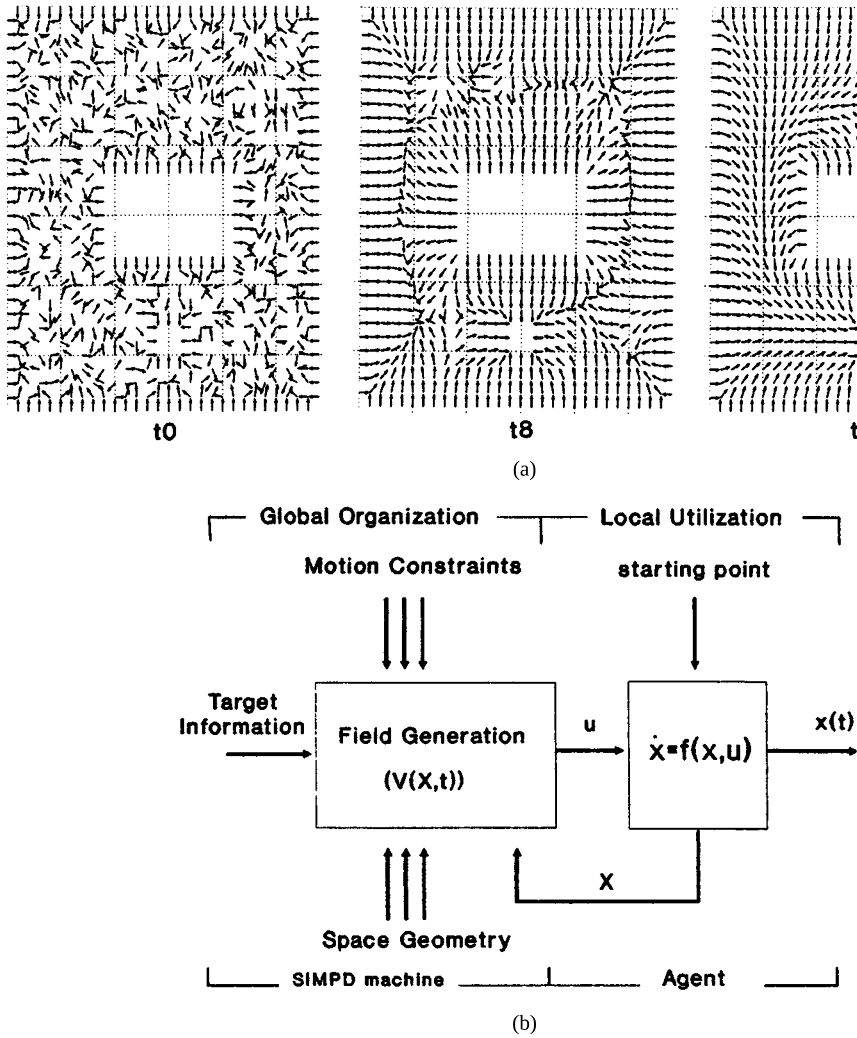


Fig. 2. (a) Evolution of a valid control group structure in a NC constructed using a hybrid, PDE-ODE, potential-based system. (b) Basic structure of a hybrid, PDE-ODE, Potential-based planning method.

at each point in state space is important for constructing a NC. It also has other advantages described in [1]–[6].

The potential field approach to motion planning is rich with techniques that can embed an agent in the context of its environment. For an extensive survey of potential-based planning methods that covers up to 1994, see Masoud [6]. To the best of these authors' knowledge, the potential approach was the first to be used for generating a paradigm for motion guidance [7], [8]. The paradigm is based on the simple idea of an attractor field situated on the target and a repeller field fencing the obstacles. Several decades later, the paradigm surfaced again through the little known work of Loef and Soni which was carried out in the early 1970's [9], [10]. Not until the mid-1980's did this approach achieve recognition in the path planning literature through the works of Khatib [11], Krogh [12], [13], Takegaki and Arimoto [14] in Japan, and Pavlov and Voronin [15] in the former Soviet Union. Andrews and Hogan also worked on the idea in the context of force control [16].

Khatib began by transforming the system equation of the manipulator

$$\mathbf{D}(\mathbf{x})\ddot{\mathbf{x}} + \mathbf{C}(\mathbf{x}, \dot{\mathbf{x}}) + \mathrm{g}(\mathbf{X}) = \mathbf{u} \tag{4}$$

into a decoupled system of unit masses using $\mathbf{u} = \mathbf{D}(\mathbf{x})\mathbf{F}^* + \mathbf{C}(\mathbf{x}, \dot{\mathbf{x}}) + \mathrm{g}(\mathbf{x})$, where $\mathbf{x}$ is an operational set of coordinates [17], $\mathbf{D}(\mathbf{X})$ is a symmetric, positive definite inertia matrix, $\mathbf{C}$ is a vector containing the coriolos and centripetal forces, g is the gravity vector, and $\mathbf{u}$ is the externally applied generalized force. The vector $\mathbf{F}^*$ is the sum of the following forces: a force $\mathbf{F}_\alpha$ that is the negative gradient flow of an attractive potential field which surrounds a reference point, a repulsion force $\mathbf{F}_r$ generated by a repulsive potential field that fences the obstacles, and a linear damping force $\mathbf{F}_d$.

Although the approach proved effective, it suffered from two problems: 1) a cluttered environment causes local minima to form which traps the manipulator before reaching its target, and 2) the potential used for the obstacles is an inverse quadratic function that may result in a high force, which causes an unrealizable control effort. Also, the interaction between $\mathbf{F}_r$ and $\mathbf{F}_\alpha$ may cause transients that are hard to control. Andrews and Hogan adopted an impedance control approach to path planning in which the environment is treated as an admittance and the manipulator as an impedance.

The approach closely resembles Khatib's method. As in Khatib's approach, trap situations due to local equilibrium

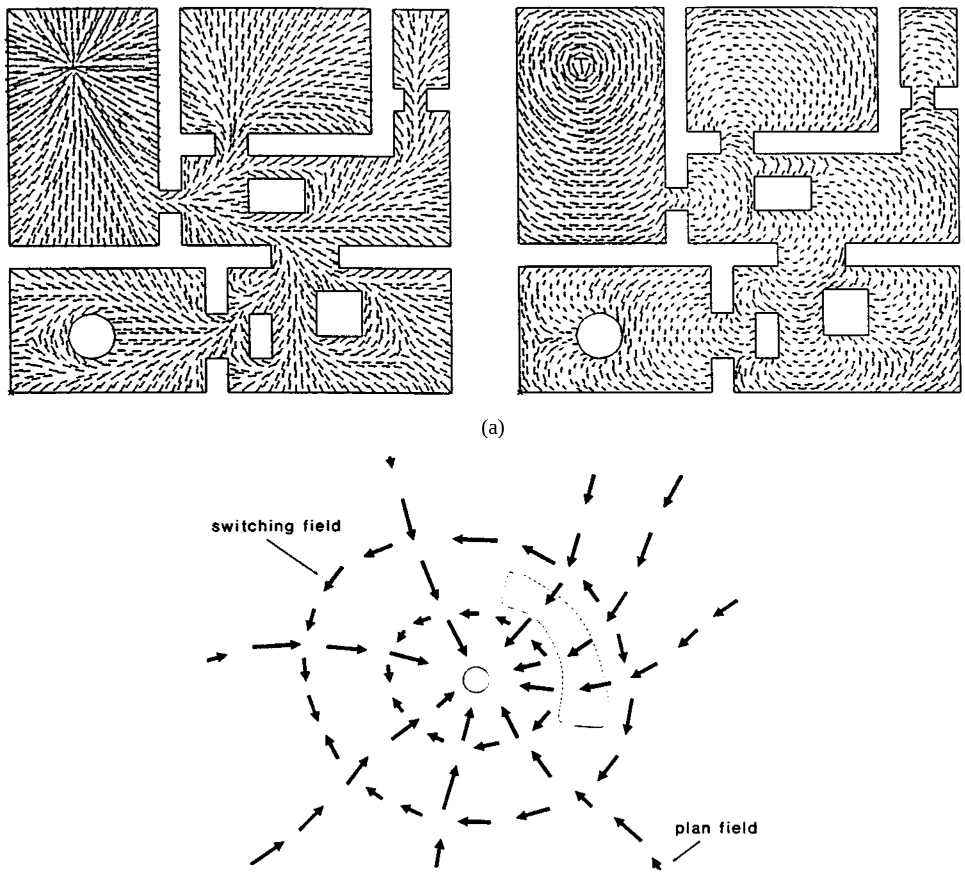


Fig. 3. (a) A general, curvilinear, boundary-fitted coordinate system from a vector potential realized by two gradient fields from scalar potentials. (b) Switching field from a VPF augmenting a plan field from a SPF.

zones were also a problem. Newman and Hogan expressed the potential approach using an energy interpretation that is similar to the time optimal bang-bang control [18]. The approach was later developed by Newman *et al.* in [79]–[81]. Takegaki and Arimoto started by deriving the system equation using the Hamiltonian. A robust feedback stabilization input $(\mathbf{u})$ to steer an arbitrary point in the unconstrained configuration space to a target point was constructed by modifying the potential energy of the system using the feedback

$$\mathbf{u} = -\left(\frac{\partial V^o}{\partial \mathbf{x}}\right)^t + \left(\frac{\partial V}{\partial \mathbf{x}}\right)^t \tag{5}$$

where V^o is a desirable potential function that is constructed in accordance with the aim of the control, and V is the potential of the system.

Krogh suggested making the strength of repulsion directly proportional to the speed of approach and inversely proportional to the minimum avoidance time [12]. He proposed that the avoidance vector for an obstacle be the gradient of a position and velocity-dependent potential field $(V(\mathbf{x}, \dot{\mathbf{x}}))$ which he referred to as the generalized potential field (GPF) [12]. In a subsequent work, Krogh approached the problem more generally: to transfer the state of a dynamical system from an initial state to a final one, avoiding undesired regions along the way [13].

Unfortunately, he did not supply a formal procedure for deriving the GPF. Also, his attempt to restrict the control fencing the obstacles to the boundary of the forbidden regions raises serious questions about the ability of a finite strength control to prevent the state from entering those regions. Tilove compared the classical potential field with the generalized potential using different utilization strategies [19]. He found that the results obtained using the generalized potential field yield a smoother trajectory that better suits the dynamics of the robot. Another comparison and a critical, empirical study of potential field methods may be found in [73] and [74] respectively.

A method for constructing a NC that bears great resemblance to a potential field method is avoidance control. Avoidance control was suggested to keep the state of a dynamical system outside a specific region in state space [20]. A refinement of Avoidance control, the optimal avoidance control (OAC), was also suggested [21]. OAC functions to maximize the minimum distance from an avoidance region while transferring the dynamical system from an initial state to a final one. More work on the subject may be found in [22]–[25]. Unfortunately, the approach faced two major stumbling blocks, halting further investigation. The first problem was its inability to provide a formal procedure for deriving the control; only guidelines were provided. Generating a form for the control was left to the subjectivity of the designer. The second and more serious difficulty was the OAC's failure to handle nonconvex regions. Even in [25], where con-

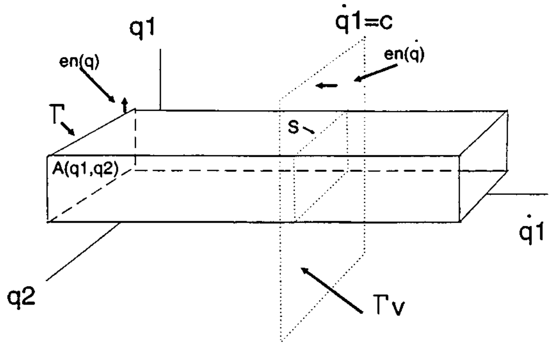


Fig. 4. Hyper-cylinders representing the velocity and position avoidance regions and their intersection S.

ditions for navigation in the presence of nonconvex avoidance regions were derived, the authors reported failure in every attempt to use these conditions for constructing a control for the nonconvex case.

In [26], Koditschek showed that it is impossible to construct a potential function with a vector field that can guarantee global convergence to a target point. However, "Almost Global" convergence is possible. Procedures were suggested for building navigation potential functions for a variety of workspaces that are geometrically different but topologically equivalent [27]–[30]. Koditschek *et al.*showed that the gradient of the potential field, with the appropriate dissipative vector field $(\mathbf{d})$ is satisfactory for constructing the navigation control [31]–[33]

$$\mathbf{u} = -\nabla v(x) + \mathbf{d}(x, \dot{x}). \tag{6}$$

Unfortunately, the control scheme does not mention how to deal with the gravity term. Therefore, these authors will assume that they relied on the troublesome cancellation strategy. Also, they imposed an initial speed limit on the robot which has to be provided as a function of the initial position. The violation of this constraint could lead to the robot penetrating an avoidance region. No method for computing this limit was provided. Sundar and Shiller combined the idea of acceleration lines with that of potential fields to achieve a near time-optimal trajectory to the target [34]. Their strategy is to augment the above technique with an acceleration potential and a deceleration potential at the terminal points of motion. This potential is designed so that it does not introduce undesirable local equilibrium, and it fades away with distance from the terminal points. The authors reported that in most of their experiments the resulting time came as close as 2% to the optimal one. However, there is no mention of the effect of initial acceleration on the collision avoidance ability of the method.

Of particular significance are potential field methods that use the flow-lines of surfaces providing solutions to certain boundary value problems. These methods can be expressed in the hybrid partial differential equation-ordinary differential equation (PDE-ODE) system format shown in Fig. 2(b). This class of planners is well suited for integrating an agent in the context of its environment, and, in turn, for constructing a NC. For a detailed discussion of this class of planners see [1]–[6]. To the best of this authors' knowledge the first such

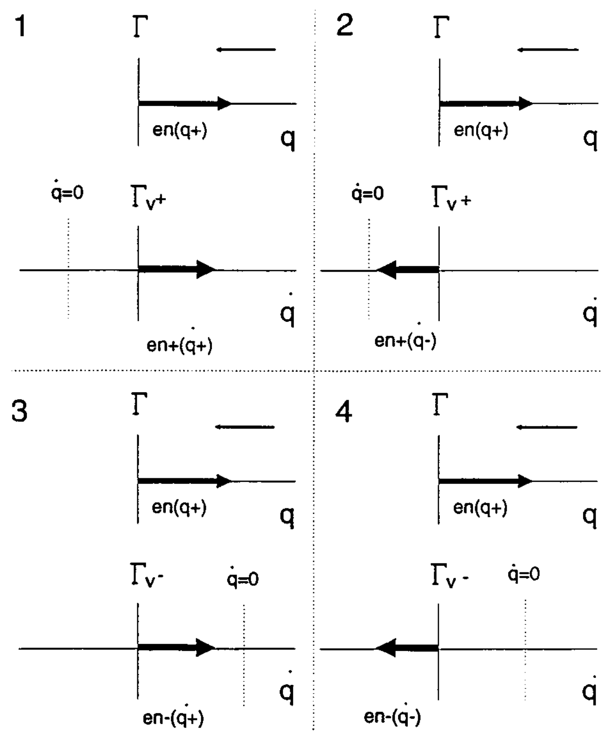


Fig. 5. Possible combinations of PPC's in the positions and velocity spaces.

method was proposed by Satoh in the mid-1980's [35]. By requiring the potential field $(V(\mathbf{x}))$ to be harmonic, thereby satisfying the Laplace equation $(\nabla^{2} V(\mathbf{x}) = 0)$, it is possible to generate a gradient field $(-\nabla V(\mathbf{x}))$ with flow-lines that mark collision-free paths to the target set. Unfortunately, because the work was published only in Japanese, it received minimal exposure. For an English version of the work, see [36].

Other methods for utilizing harmonic potential fields in motion planning were later suggested in [37]–[51]. Biharmonic potential field techniques ($V(\mathbf{x})$ satisfies $\nabla^{4} V(\mathbf{x}) = 0$) were found to favorably compare to their harmonic counterparts by producing paths with lower curvature and potential fields that can be reliably computed for workspaces with excessively complex geometry [52]. Furthermore, techniques based on potential fields that satisfy the diffusion equation [53] or the wave equation [54] were suggested for motion planning for nonstationary targets. Unfortunately, the above techniques only mark a safe path to a target set.

Additional conditioning is required to convert the guidance signal that such potential fields provide into a control signal that would instruct the robot to properly deploy its actuators of motion, enabling the target to be safely reached. An interesting approach for generating a NC signal from a guidance field signal was suggested by Utkin *et al.* [55]–[59]. The approach utilizes the sliding mode theory to force the state of the robot to track the lines of the guidance field. The authors applied their approach for the special case of a gradient guidance field. However, the approach is so general that any type of guidance field could be accommodated. Also, for the control effort to be finite, the lines of ∇V must have bounded curvature. Other procedures for converting the guidance field from a harmonic potential to a NC signal may be found in [60] and [61].

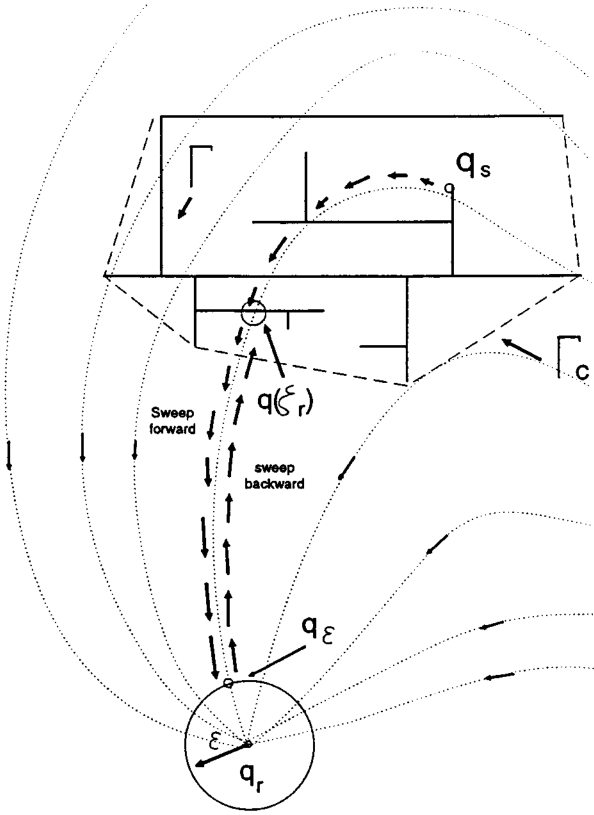


Fig. 6. Computing an exit point on the surface of an obstacle.

In [75], a potential function is treated as a Liapunov function and is used in real-time to derive a control signal for the constrained proximity maneuvering of a low-earth-orbit space platform. The maneuver consists of driving the platform to a rendezvous position while avoiding a convex obstruction region. Potential fields were also used for designing impact controllers to tackle the problem of real-time, collision-free motion of a space vehicle through an environment, reduced velocity of approach of surfaces to be contacted or docked with, and force control [76]. In [84] potential shaping and dissipation are employed to obtain full exponential stabilization to a desired trajectory of a mechanical system.

Despite the variety of methods that were proposed for building a NC, there is still a strong demand for constructing NC's that can satisfactorily control the quality of behavior and provide strict guarantees that practical behavioral constraints can be imposed and satisfied. Most importantly is a demand to yield a flexible control signal so that the amount of change to the constraints on behavior is commensurate with the effort needed to adjust the NC. This paper presents an attempt to attain such qualities in a NC. The suggested approach heavily relies on Vector potential fields (VPF's) for inducing the control action. VPF's fundamentally differ in nature from scalar potential fields (SPF's) which, to the best of these authors' knowledge, have previously been the only kind of potential fields used for synthesizing NC's.

In Section II, the need to use a VPF to generate a NC instead of a SPF is discussed. A strategy for navigation is suggested in Section III. Sections IV and V discuss NC generation. Section VI introduces nonlinear, anisotropic motion damping. In Section VII two examples are supplied to demonstrate the capabilities of the proposed method. Conclusions are presented in Section VIII.

II. Why a Vector Potential

Projecting an action that satisfies the goal and upholds the constraints on behavior requires the generation of a sequence of control signals $(\mathbf{u_1}, \mathbf{u_2}, \ldots, \mathbf{u}_l)$ that yield a corresponding sequence of states $(\mathbf{x_1}, \mathbf{x_2}, \ldots \mathbf{x}_l)$ so that the final state $(\mathbf{x}_l)$ is the desired target state and all the transient states satisfy the constraints on behavior. Such a sequence is called a plan. In a potential field approach, such a plan is a member of a field of plans (action field) that densely covers state space, so that regardless of the starting point $(\mathbf{x_0})$, a plan always exists to safely propel the robot to its destination. SPF methods, especially those that utilize the potential field in the context of a Hybrid PDE-ODE system, were proven to be efficient tools for generating the above capabilities. Fig. 3(a) shows the action field for the simple drift-free system $\dot{\mathbf{x}} = \mathbf{u}$, generated by the method in [52].

Such methods require a model of the environment that is known *a priori* to be able to generate the NC. Unfortunately, a realistic changing environment significantly shortens the life of any *a priori* nown odel. This may render part or all of the plans which the action field encodes to be invalid. What makes the plans generated by a SPF particularly susceptible to changes in the environment is the fact that each starting point in state space defines one and only one plan to the target. If the plan fails, the robot needs to recompute the whole action field taking into account the new information about the workspace in order to generate a new valid plan. This is a considerable burden, particularly when considering a multidimensional workspace.

Instead of recomputing the whole action field, it is more reasonable for the robot to attempt to ameliorate utilization by switching from the failed plan to a valid one which, when found, can safely guide the robot to its target. Unfortunately, SPF techniques are inherently incapable of functioning in such a manner. SPF techniques generate the NC field from the gradient flow $(\nabla(\;))$ of a surface that is either the potential itself, or a scalar function $(S(\;))$ of that potential

$$\mathbf{u} = -\nabla(S(V(\mathbf{x}))) \tag{7}$$

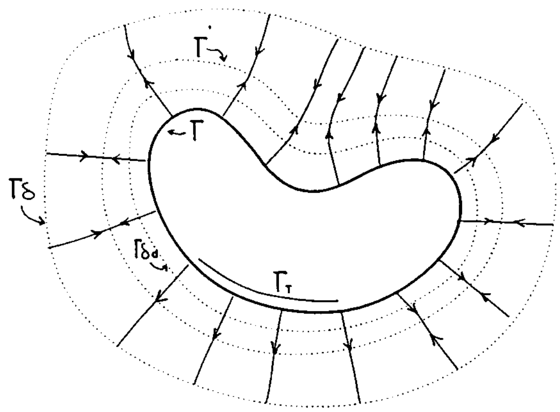
(e.g., the control may be $\mathbf{u} = -\nabla V$ or, as in [52], $\mathbf{u} = -\nabla(\nabla^{\mathbf{2}} V)$.)

It is well known that the gradient flow of a surface degenerates along the family of equipotential contours [tangent space of the surface which is orthogonal to the gradient flow (normal space of the surface)]. This may be deduced from the vector identity

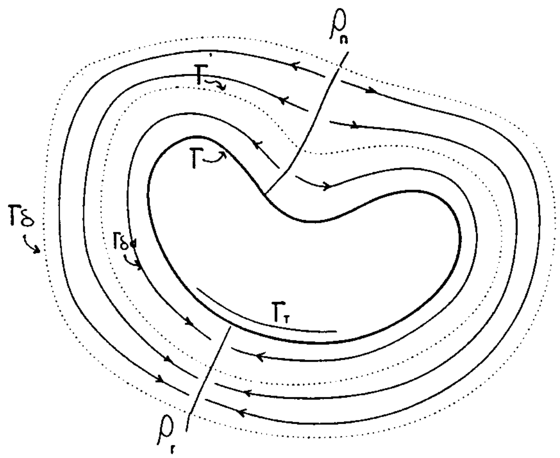
$$\nabla_X(\nabla S(V(\mathbf{x}))) \equiv 0 \tag{8}$$

where ∇_X is the curl operator which is used to detect the circulating field along the tangent space.

A SPF control field is incapable of driving motion along a trajectory orthogonal to the gradient flow lines. Therefore, a SPF NC is incapable of switching between plans, confining the mo-



(a)



(b)

Fig. 7. (a PPC component. (b) LAC component.

tion of the state to one and only one solution trajectory. A SPF gradient field has no control over motion in the tangent space. With the loss of controllability over the tangent space, known to span $N-1$ degrees of freedom in an N-dimensional space, the effectiveness of the gradient field (normal space control component) in steering motion seriously deteriorates with an increase in the space dimensionality. To remedy this shortcoming, vector potential fields $(\mathbf{V}(\mathbf{x}))$ are suggested. VPF's are able to synthesize a complete set of basis vector fields that may be used to construct a control that has better ability to steer a robot in its workspace. To see the relevance to the plan switching problem described above, Helmholtz's theorem is used to partition the control action from a general vector potential field $(\mathbf{V})$ into two functionally distinct components ([62, vol. 1, p. 52]). The first component is a conservative gradient field of a scalar potential that functions as the action field of the robot. The second component is generated from the curl of a constrained vector potential to play the role of the tangential switching field circulating the equipotential surfaces of the action field. Helmholtz's theorem is stated below with minor changes to the notation.

Theorem: Any vector field $\mathbf{u}$ that is finite, uniform, vanishes at infinity, and continuous may be expressed as the sum of a gradient scalar field and the curl of zero-divergence vector field

$$\mathbf{u} = -\nabla S(V(\mathbf{x})) + \nabla X \mathbf{A}(\mathbf{x}), \quad \nabla \cdot \mathbf{A}(\mathbf{x}) \equiv 0 \tag{9}$$

where V is the scalar potential of $\mathbf{u}$, $\mathbf{A}$ being its vector potential, $\nabla \cdot$ is the divergence operator

$$\nabla X \mathbf{A}(\mathbf{x}) = \begin{bmatrix} 0 & -\partial/\partial z & \partial/\partial y \\ \partial/\partial z & 0 & -\partial/\partial x \\ -\partial/\partial y & \partial/\partial x & 0 \end{bmatrix} \begin{bmatrix} A_x(\mathbf{x}) \\ A_y(\mathbf{x}) \\ A_z(\mathbf{x}) \end{bmatrix}$$

has a purely circulating nature, $\mathbf{A}(\mathbf{x}) = [Ax(\mathbf{x})Ay(\mathbf{x})Az(\mathbf{x})]t$, and $\mathbf{x} = [x \;\; y \;\; z]^t$. Since the control component is intended for use as a local modifier of the preexisting, global control, the restrictions for partitioning $\mathbf{u}$ are applicable. As illustrated, the control field from a VPF does accommodate the action field from a SPF. Moreover, it provides the robot with the option of switching from one plan to another if needed [Fig. 3(b)].

The underlying potential field from which a modifying control action is generated may be derived by solving a properly formulated boundary value problem (BVP.) Formulating a BVP requires

a) a partial differential relation to govern the differential properties of the field;
b) boundary conditions (in the sequel, boundary conditions are called boundary control).

The governing partial differential relation should be selected to guarantee the ability of the local control field to modify the preexisting, global control component. Synthesizing the conservative gradient term of (9), Action Field, can be carried out with no difficulties in a multidimensional space. Unfortunately, this is not so for the curl component of the control. While a definition of the curl operator exists in two-dimensional, three-dimensional, and four-dimensional spaces ([83], p. 135), the authors were not able to find a general definition for this operator in N-dimensional spaces. Since there is no proof that the operator cannot be defined for higher dimensional spaces, its existence is assumed along with the ability to synthesize a control action from $\mathbf{A}$ in N-dimensional spaces. Thus, the authors suggest a general method for realizing a control component from the vector potential while bypassing the need to have an explicit definition of the curl operator. The suggested procedure is inspired by the Gram–Schmidt orthogonalization method [82]. This method is used to convert a set of vectors into an orthogonal one. Gram–Schmidt method begins by arbitrarily selecting a member from the set that is to be orthogonalized as the first vector in the orthogonal set. Therefore we will begin by selecting the Action Field $-S(V(\mathbf{x}))$ as the first component in the orthogonal set of basis vector fields used to construct $\mathbf{u}$. As will be shown later, the conservative component $(-\nabla V, S(\mathbf{x}) = \mathbf{x})$ may be generated by solving the BVP

$$\nabla \cdot \nabla V = \nabla^2 V \equiv 0. \tag{10}$$

Subject to the proper set of boundary conditions (BC). Although $\mathbf{A} \in R^N$, due to the auxiliary condition $\nabla \cdot \mathbf{A} \equiv 0$, the independent scalar quantities needed for completely specifying $\mathbf{A}$ drop from N to $N-1$. Therefore $N-1$ scalar potential fields $(V_i : i = 1, \ldots, N-1)$ are needed to represent $\mathbf{A}$. V_1 may be generated by solving the BVP

$$\nabla^2 V_1 \equiv 0 \tag{11}$$

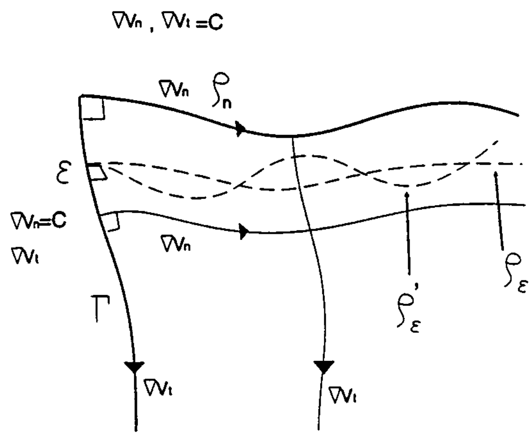


Fig. 8. Field lines near the boundary.

subject to the first set of boundary conditions (BC1). BC1 has to be chosen so that

$$\nabla V_{\mathbf{1}}^t \nabla V \equiv 0$$

$V_{\mathbf{2}}$ is generated by solving the BVP

$$\nabla^{\mathbf{2}} V_{\mathbf{2}} \equiv 0$$

subject to the second set of boundary conditions (BC2). BC2 has to be chosen so that

$$\nabla V_{\mathbf{2}}^t \nabla V \equiv 0, \quad \text{and} \quad \nabla V_{\mathbf{2}}^t \nabla V_{\mathbf{1}} \equiv 0.$$

The above is continued till all the $N-1$ scalar potentials are computed. The control action from $\mathbf{A}$ is constructed as

$$\nabla X \mathbf{A}(X) = \sum_{i=\mathbf{1}}^{N-\mathbf{1}} -\nabla V_i(\mathbf{x}) \tag{12}$$

where it is required that

$$\nabla S(V(\mathbf{x}))^t \nabla V_i(\mathbf{x}) \equiv 0 \quad i = 1, \ldots, N-1$$

and

$$\nabla V_i(\mathbf{x})^t \nabla V_j(\mathbf{x}) \equiv 0 \quad i \neq j.$$

The above procedure is equivalent to the parameterization of space using fitted, general curvilinear coordinate systems [Fig. 3(a)]. It is shown in the sequel that only one out of the $N-1$ scalar potential field components is needed to construct a control action from $\mathbf{A}$ ($\nabla \times \mathbf{A} = -\nabla V1$) that can successfully steer the state toward valid solution trajectories.

III. The Proposed Navigation Strategy

To achieve the flexibility that is desired in a NC, a decentralized approach is used to accommodate the presence of new obstacles in the navigation process. Here, the existing NC ($\mathbf{u}_g$) is augmented with local, noninteracting control component ($\mathbf{u}_l$) to steer the state away from old solution trajectories (trajectories formed by $\mathbf{u}_g$) that violate the newly introduced constraints to old solution trajectories that are still valid. The new NC has the form

$$\mathbf{u} = \mathbf{u}_g(\mathbf{q}) + \mathbf{u}_l(\mathbf{q}, \mathbf{u}_g). \tag{13}$$

$\mathbf{u}_g$ functions as a NC for the pre-existing obstacles (O_{old}) such that for

$$\mathbf{D}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{c}(\mathbf{q}, \dot{\mathbf{q}}) + \mathbf{g}(\mathbf{q}) = \mathbf{u} \tag{14}$$

$$\lim \mathbf{q}(t) \to \mathbf{q}_r \quad t \to \infty$$

and

$$\mathbf{q}(t) \cap O_{\text{old}} = \phi \quad \text{for all } t$$

where $\mathbf{q}(t)$, $\dot{\mathbf{q}}(t)$, $\ddot{\mathbf{q}}(t)$ all $\in R^N$ are the position, velocity, and acceleration vectors of the robot in its natural coordinates, ϕ is the empty set, and $\mathbf{q}_r$ is the desired target point in the workspace. Note that O_{old} may be the empty set ϕ, and $\mathbf{u}_g$ may be as simple as a Proportional-Derivative (PD) controller [63]

$$\mathbf{u}_g = \mathbf{K} \cdot (\mathbf{q} - \mathbf{q}_r) + \mathbf{B} \cdot \dot{\mathbf{q}} \tag{15}$$

where $\mathbf{K}$ and $\mathbf{B}$ are $N \times N$ positive definite matrices. On the other hand, $\mathbf{u}_l$ handles the newly introduced set of obstacles O ($\Gamma = \partial O$). The new obstacles (O_{new}) are the union of the sets O_{old} and O

$$O_{\text{new}} = O_{\text{old}} \cup O. \tag{16}$$

The local steering control ($\mathbf{u}_l$) is strictly localized to the vicinity of O. It is designed so that $\mathbf{u}$ is able to make the system in (14) satisfy

$$\lim \mathbf{q}(t) \to \mathbf{q}_r \quad t \to \infty$$

and

$$\mathbf{q}(t) \cap O_{\text{new}} = \phi \quad \text{for all } t. \tag{17}$$

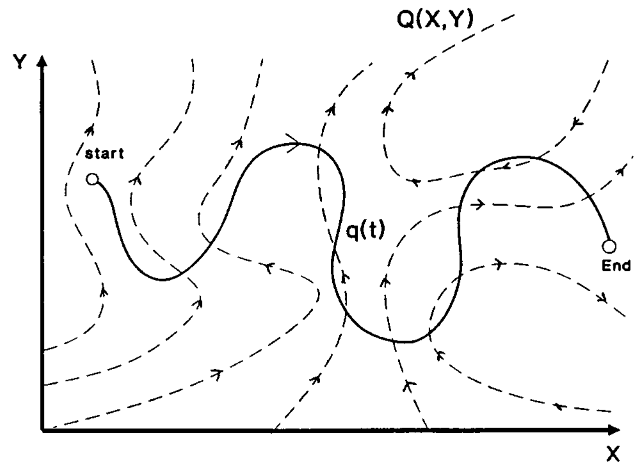


Fig. 9. Motion damping along the lines of Q.

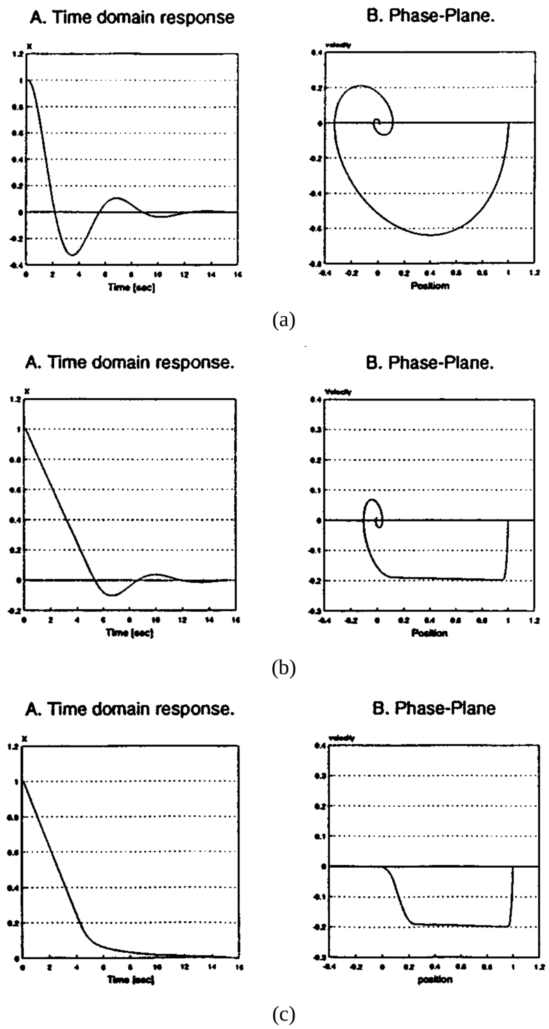


Fig. 10 (a) Unconstrained system, ζ. (b) Velocity constrained not to go below -0.2. (c) Velocity constrained not to go below -0.2, and position constrained not to be go below 0.0.

The local component $(\mathbf{u}_l)$ is designed in the local coordinates of the obstacles, then transformed to the natural coordinates of the robot. It is divided into the two functionally distinct components $\mathbf{u}_{ln}$ and $\mathbf{u}_{lt}$

$$\mathbf{u}_l = \mathbf{u}_{ln} + \mathbf{u}_{lt}. \tag{18}$$

The first $(\mathbf{u}_{ln})$ is called the penetration prevention control (PPC.) It acts normally to the surface of O to prevent the robot from penetrating that region. The other component $(\mathbf{u}_{lt})$ is called the local alignment control (LAC). This component acts tangentially to the surface of O in order to drive the robot to a proper position on Γ where $\mathbf{u}_g$ can assume command of the navigation process and sweep the robot to $\mathbf{q}_r$.

The steering control $(\mathbf{u}_l)$ is made to occupy a small neighborhood $(O\delta_d)$ around O that has a minimum width of δ_d and a boundary $\Gamma\delta_d(\Gamma\delta_d = \partial O\delta_d)$. The added region is needed to guarantee that $\mathbf{u}_l$ is bounded. As will be shown in the sequel, δ_d is inversely proportional to the magnitude of $\mathbf{u}_l$ (see Example 1). For a smooth diversion of motion away from the obstacles, $\mathbf{u}_l$ is made to gradually decay to zero in a finite region $O\delta$ surrounding $O\delta d$. The control on the inner boundary of $O\delta$ $(\Gamma\delta_d = \partial O\delta)$ is derived in the following sections, while the control on and outside the outer boundary $(\Gamma\delta)$ is set to zero. The steering control inside $O\delta$ is generated by solving a vector boundary value problem (VBVP) which, for convenience, is replaced by the solution of, at the most, four scalar boundary value problems (at the most two for each component of $\mathbf{u}_l$ one to construct the phase control field and the other to construct the magnitude field). The rest of this paper discusses the construction of $\mathbf{u}_l$.

IV. The Boundary Steering Control

In this section the steering control at the boundary of O $(\mathbf{u}_{\Gamma l})$ is derived for later use in generating $\mathbf{u}_l$. The boundary control is derived in terms of the obstacle's normal and tangent basis vectors $\mathrm{e}_n(\mathbf{q})$ and $\mathrm{e}_t(\mathbf{q})$ respectively. In the next section, the generation of these vectors along with the steering control are discussed. Similar to $\mathbf{u}_l$, $\mathbf{u}_{\Gamma l}$ has two components: the Boundary Penetration Prevention Control (BPPC, $\mathbf{u}_{\Gamma ln}$), and the Boundary Local Alignment Control (BLAC, $\mathbf{u}_{\Gamma lt}$).

A. The BPPC in the Position Space

Proposition-1: For a control law of the form

$$\mathbf{u}_{\Gamma_{ln}}(\mathbf{q}, \dot{\mathbf{q}}) = \alpha_{\mathbf{1}}(\mathbf{q}, \dot{\mathbf{q}})\mathrm{e}_n(\mathbf{q}) \quad q \in O\delta_d$$

where

$$\alpha_{\mathbf{1}}(\mathbf{q}, \dot{\mathbf{q}}) = \alpha_{\mathbf{11}}(\mathbf{q}, \dot{\mathbf{q}}) + k \cdot |\dot{\mathbf{q}}_n| \tag{19}$$

$\alpha_{\mathbf{11}}$ is a scalar function, k is a constant, and $\dot{\mathbf{q}}_n = \dot{\mathbf{q}}^t \mathrm{e}_n(\mathbf{q})$, there exist an $\alpha_{\mathbf{11}} > 0$, and a $k > 0$ such that the above control prevents $\mathbf{q}$ in (14) from entering O

$$\mathbf{q} \cap O = \phi \quad \text{for all } t. \tag{20}$$

Proof: Let q_n be the normal distance from Γ to the present location of the robot:

$$q_n = \mathbf{q}^t \mathbf{e}_n(\mathbf{q}). \tag{21}$$

Let G be the distance measure

$$G = 1/2q_n^{\mathbf{2}}. \tag{22}$$

In the following, it is shown that there is a choice for k and $\alpha_{\mathbf{11}}$ that makes the time derivative of G greater than or equal to zero

$$\dot{G} = q_n \dot{q}_n = q_n[\mathrm{e}^t(\mathbf{q})\dot{\mathbf{q}}] \geq 0.$$

This is sufficient to prove that the robot will never touch Γ. Since, initially, $\mathbf{q} \notin 0$ (i.e. $q_n > 0$), the condition for guaranteeing that $\dot{G} \geq 0$ is reduced to guaranteeing that

$$\mathbf{e}_n^t(\mathbf{q})\dot{\mathbf{q}} \geq 0. \tag{23}$$

The speed at time t may be represented as

$$\dot{\mathbf{q}}(t) = \int_{t\mathbf{0}}^{t} \ddot{\mathbf{q}}(t)\, dt. \tag{24}$$

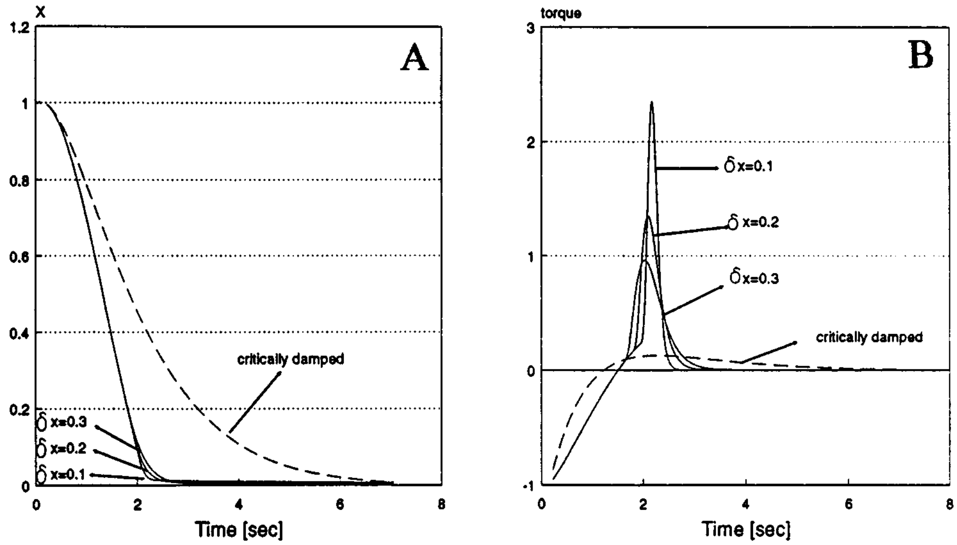


Fig. 11. (a) Response to different values of δ_x, (b). corresponding forces.

Assuming that motion starts outside Γ, and at the present time is inside $\Gamma\delta_d$, the above integral can be divided into two parts:

$$\begin{aligned}\dot{\mathbf{q}}(t) &= \int_{t0}^{t^-} \ddot{\mathbf{q}}(t)\,dt + \int_{t^-}^{t} \ddot{\mathbf{q}}(t)\,dt \\ &= \dot{\mathbf{q}}^- + \int_{t^-}^{t} [\mathbf{f}(\mathbf{q},\dot{\mathbf{q}}) + \mathbf{D}^{-1}(\mathbf{q})\mathbf{u}]\,dt \end{aligned} \tag{25}$$

where $\mathbf{f}(\mathbf{q},\dot{\mathbf{q}}) = -\mathbf{D^{-1}}(\mathbf{q})[\mathbf{C}(\mathbf{q},\dot{\mathbf{q}}) + \mathrm{g}(\mathbf{q}) - \mathbf{u}_g]$, t0 is the time at which motion starts, t is the present time, t^- is the time at which the robot enters O_{δ_d}, and $\dot{\mathbf{q}}^-$ is the corresponding entrance speed ($\dot{\mathbf{q}}$ at $\Gamma\delta_d$). Substituting the speed in (23) we have

$$\left[\mathrm{e}_n^t(\mathbf{q})\left[\dot{\mathbf{q}}^- + \int_{t^-}^{t} [\mathbf{f}(\mathbf{q},\dot{\mathbf{q}}) + \mathbf{D}^{-1}(\mathbf{q})\mathbf{u}]\,dt\right]\right]. \tag{26}$$

For the above to be greater than or equal to zero, one must have

$$\int_{t^-}^{t} \mathrm{e}_n^t(\mathbf{q})[\mathbf{f}(\mathbf{q},\dot{\mathbf{q}}) + \mathbf{D}^{-1}(\mathbf{q})\mathbf{u}]\,dt \geq |\dot{\mathbf{q}}_n^-| \tag{27}$$

where $\dot{\mathbf{q}}_n^- = \mathrm{e}_n^t\dot{\mathbf{q}}^-$ (note that e_n is constant with respect to time). Substituting $\mathbf{u}_{\Gamma In}$ in (27), we have

$$\begin{aligned}\int_{t^-}^{t} &\mathrm{e}_n^t(\mathbf{q})[\mathbf{f}(\mathbf{q},\dot{\mathbf{q}}) + (\alpha_{\mathbf{11}}(\mathbf{q},\dot{\mathbf{q}}) \\ &+ k\cdot|\dot{\mathbf{q}}_n|)\mathbf{D^{-1}}(\mathbf{q})\mathrm{e}_n(\mathbf{q})]\,dt \geq |\dot{\mathbf{q}}_n^-|. \end{aligned} \tag{28}$$

One way to guarantee that the above inequality holds is to require that

$$\int_{t^-}^{t} \left[\mathrm{e}_n^t(\mathbf{q})\mathbf{f}(\mathbf{q},\dot{\mathbf{q}}) + \alpha_{\mathbf{11}}(\mathbf{q},\dot{\mathbf{q}})\mathrm{e}_n^t(\mathbf{q})\mathbf{D^{-1}}(q)\mathrm{e}_n(\mathbf{q})\right]dt \geq 0 \tag{29}$$

and

$$\int_{t^-}^{t} k\cdot|\dot{\mathbf{q}}_n|\mathrm{e}_n^t(\mathbf{q})\mathbf{D^{-1}}(\mathbf{q})\mathrm{e}_n(\mathbf{q})\,dt \geq |\dot{\mathbf{q}}_n^-|. \tag{30}$$

The first term guarantees that the robot will not be accelerated inside $O\delta_d$, while the second term guarantees that the kinetic energy acquired prior to the robot entering $O\delta_d$ is dissipated. The independent nature of the two terms is a good indicator that such a division does not lead to a conservative control law. Since an integral with a positive argument is positive, the first condition can be enforced by choosing

$$\operatorname*{Inf}_{\mathbf{q},\dot{\mathbf{q}}} \alpha_{\mathbf{11}}(\mathbf{q},\dot{\mathbf{q}}) \geq \frac{\mathrm{Sup}_{\mathbf{q},\dot{\mathbf{q}}}|\mathrm{e}_n^t(\mathbf{q})\mathbf{f}(\mathbf{q},\dot{\mathbf{q}})|}{\mathrm{Inf}_{\mathbf{q}}\left(\mathrm{e}_n^t(\mathbf{q})\mathbf{D}^{-1}(\mathbf{q})\mathrm{e}_n(\mathbf{q})\right)}. \tag{31}$$

Notice that since $\mathbf{D^{-1}}(\mathbf{q})$ is positive definite and $\mathrm{e}_n(\mathbf{q})$ is a basis vector (i.e., it cannot be zero), the denominator is always greater than zero. As for the second term, k is computed by first constructing the following lower bound on the integral in (30)

$$\begin{aligned}&\int_{t^-}^{t} \left[k\cdot|\dot{\mathbf{q}}_n|\mathrm{e}_n^t(\mathbf{q})\mathbf{D^{-1}}(\mathbf{q})\mathrm{e}_n(\mathbf{q})\right]dt \\ &\geq \min_{\mathbf{q}}\left(\mathrm{e}_n^t(\mathbf{q})\mathbf{D}(\mathbf{q})\mathrm{e}_n(\mathbf{q})\right)\int_{t^-}^{t} k\cdot|\dot{\mathbf{q}}_n|\,dt \\ &= k\cdot\min_{\mathbf{q}}\left(\mathrm{e}_n^t(\mathbf{q})\mathbf{D}^{-1}(\mathbf{q})\mathrm{e}_n(\mathbf{q})\right)|q_n(t_f) - q_n(t^-)|. \end{aligned}$$

Using the inequality $|q_n(t_f) - q_n(t^-)| \geq \delta_d$ to further bound the above, we have the integral bounded by

$$\geq k\cdot\delta_d\left[\min_{\mathbf{q}}\left(\mathrm{e}_n^t(\mathbf{q})\mathbf{D}^{-1}(\mathbf{q})\mathrm{e}_n(\mathbf{q})\right)\right] \tag{32}$$

where t_f is the time the force field stops the robot short of hitting the obstacle. Therefore, the second condition can be guaranteed if

$$k\cdot\delta_d\left[\min_{\mathbf{q}}\left(\mathrm{e}_n^t(\mathbf{q})\mathbf{D}^{-1}(\mathbf{q})\mathrm{e}_n(\mathbf{q})\right)\right] \geq |\dot{\mathbf{q}}_n^-| \tag{33}$$

or, equivalently

$$k \geq \frac{|\dot{\mathbf{q}}_n|}{\delta_d}\cdot\frac{1}{\min_{\mathbf{q}}\left(\mathrm{e}_n^t(\mathbf{q})\mathbf{D}^{-1}(\mathbf{q})\mathrm{e}_n(\mathbf{q})\right)}. \tag{34}$$

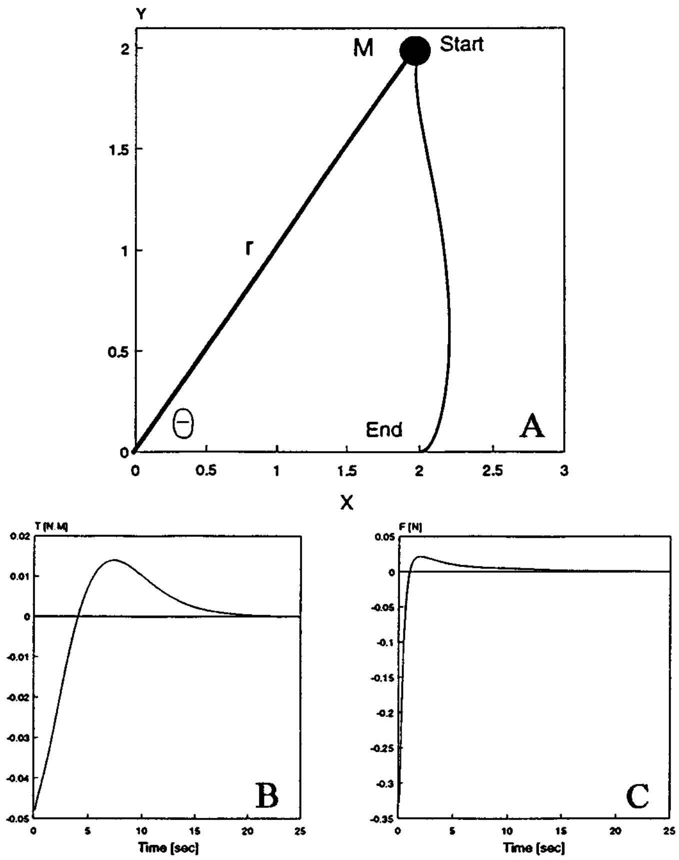


Fig. 12. (a) Robot trajectory in free space, (b) corresponding torque, and (c) corresponding force.

B. The BPPC in the Velocity Space

Proposition-2: For a control law of the form

$$\mathbf{u}_{\Gamma ln}(\mathbf{q},\dot{\mathbf{q}}) = \alpha_{\mathbf{2}}(\mathbf{q},\dot{\mathbf{q}})\mathbf{e}_n(\dot{\mathbf{q}}) \quad \mathbf{q} \in \Gamma_v \tag{35}$$

where $\alpha_{\mathbf{2}}$ is a scalar function, Γ_v is the boundary of an avoidance region in the velocity space (O_v), and $\mathbf{e}_n(\dot{\mathbf{q}})$ is a normal basis vector exactly placed on Γ_v, there exist an $\alpha_{\mathbf{2}} > 0$ so that the above control prevents $\dot{\mathbf{q}}$ of the system in (14) from entering O_v

$$\dot{\mathbf{q}} \cap O_v = \phi \quad \text{for all } t. \tag{36}$$

Note that unlike the BPPC in the position space which requires a finite region of minimum width δ_d around O in order to be realizable, a bounded BPPC in the velocity space placed exactly on Γ_v can stop $\dot{\mathbf{q}}$ from entering O_v.

Proof: Let $\dot{q_n}$ be the normal distance between $\dot{\mathbf{q}}$ and Γ_v in the velocity space

$$\dot{q}_n = \mathbf{e}_n^t(\dot{q}_n)\dot{\mathbf{q}} \tag{37}$$

and G be the distance measure

$$G = \mathbf{1/2} \cdot \dot{q}_n^{\mathbf{2}}. \tag{38}$$

To guarantee that $\dot{\mathbf{q}}$ will not enter O_v, it must be shown that for the BPPC in (35) the time derivative of G is always nonnegative

$$\dot{G}(\dot{q}_n) = \dot{q}_n\ddot{q}_n = \dot{q}_n\left[\mathbf{e}_n^t(\dot{\mathbf{q}})\cdot\ddot{\mathbf{q}}\right] \geq 0. \tag{39}$$

With the assumption that the initial velocity of the manipulator is outside O_v (i.e., $\dot{q}_n > 0$), the condition for making $\dot{G}$ nonnegative reduces to

$$\mathbf{e}_n^t(\dot{\mathbf{q}})\cdot\ddot{\mathbf{q}} \geq 0. \tag{40}$$

Substituting for $\ddot{\mathbf{q}}$ we have

$$\begin{aligned}&\mathbf{e}_n^t(\dot{\mathbf{q}})\left[\mathbf{f}(\mathbf{q},\dot{\mathbf{q}}) + \mathbf{D^{-1}}(\mathbf{q})\mathbf{u}\Gamma_{ln}\right]\\ &\quad = \left[\mathbf{e}_n(\dot{\mathbf{q}})^t\cdot\mathbf{f}(\mathbf{q},\dot{\mathbf{q}}) + \alpha_{\mathbf{2}}(\mathbf{q},\dot{\mathbf{q}})\cdot\mathbf{e}_n^t(\dot{\mathbf{q}})\mathbf{D^{-1}}(\mathbf{q})\mathbf{e}_n(\dot{\mathbf{q}})\right] \geq 0.\end{aligned} \tag{41}$$

Since $\mathbf{D^{-1}}(\mathbf{q})$ is positive definite and $\mathbf{e}_n(\dot{\mathbf{q}})$ is a basis vector (i.e., it cannot be zero), the following choice of $\alpha_{\mathbf{2}}$ guarantees that $\dot{\mathbf{q}} \cap O_v = \phi$ for all t

$$\alpha_{\mathbf{2}}(\mathbf{q},\dot{\mathbf{q}}) \geq \frac{\mathrm{Sup}_{\dot{\mathbf{q}},\mathbf{q}}\left|\mathbf{e}_n^t(\dot{\mathbf{q}})\mathbf{f}(\mathbf{q},\dot{\mathbf{q}})\right|}{\mathrm{Inf}_{\dot{\mathbf{q}},\mathbf{q}}\left(\mathbf{e}_n^t(\dot{\mathbf{q}})D^{\mathbf{-1}}(\mathbf{q})\mathbf{e}_n^t(\dot{\mathbf{q}})\right)}. \tag{42}$$

C. The Combined Position-Velocity Space

A situation may be contemplated where, in addition to avoiding the obstacles, the speed of the robot is required not to exceed a certain value. This requires the BPPC to act in both the position and velocity spaces

$$\mathbf{u}\Gamma_{ln}(\mathbf{q},\dot{\mathbf{q}}) = \alpha_1(\mathbf{q},\dot{\mathbf{q}})\cdot\mathbf{e}_n(\mathbf{q}) + \alpha_2(\mathbf{q},\dot{\mathbf{q}})\cdot\mathbf{e}_n(\dot{\mathbf{q}}). \tag{43}$$

1) Joint Constraints: Since in the state space representation a surface specified in the position space is orthogonal to that specified in the velocity space, we have

$$\alpha_1(\mathbf{q},\dot{\mathbf{q}})\cdot\alpha_2(\mathbf{q},\dot{\mathbf{q}}) = 0 \tag{44}$$

when

$$\begin{aligned}&\mathbf{q} \notin O\delta_d \quad \text{and} \quad \dot{\mathbf{q}} \notin \Gamma_v\\ &\mathbf{q} \in O\delta_d \quad \text{and} \quad \dot{\mathbf{q}} \notin \Gamma_v\\ &\mathbf{q} \notin O\delta_d \quad \text{and} \quad \dot{\mathbf{q}} \in \Gamma_v.\end{aligned}$$

However, when

$$\mathbf{q} \notin O\delta_d \quad \text{and} \quad \dot{\mathbf{q}} \in \Gamma_v$$

the above product is not zero. Such a condition occurs at the intersection of the two hypercylinders which are the extension of both $O\delta_d$ and Γ_v along $\dot{\mathbf{q}}$ and $\mathbf{q}$ respectively (Fig. 4). As a result of $\mathbf{q}$ and $\dot{\mathbf{q}}$ being related by

$$\dot{\mathbf{q}} = \frac{d\mathbf{q}(t)}{dt} = \frac{\mathbf{q}(t) - \mathbf{q}(t-dt)}{dt} \tag{45}$$

and $\mathbf{u}\Gamma_{ln}$ simultaneously actuating motion in both the velocity and position spaces, a conflict may arise between the avoidance specifications in both the velocity and position spaces. In the following, the relation between the avoidance forces from both spaces is studied for the one-dimensional case at the intersection of the two surfaces. Based on this analysis, restrictions on the N-dimensional case are deduced.

Let Γ be a contour point on the q axis, $\mathbf{e}_n(q+)$ and $e_n(q-)$ are pointing in the positive and negative directions of q respectively. Also, let Γ_{v+}, and Γ_{v-} be point contours on the positive and negative parts of the $\dot{q}$ axis, respectively. Let $\mathbf{e}_n + (\dot{q}+)$ be a

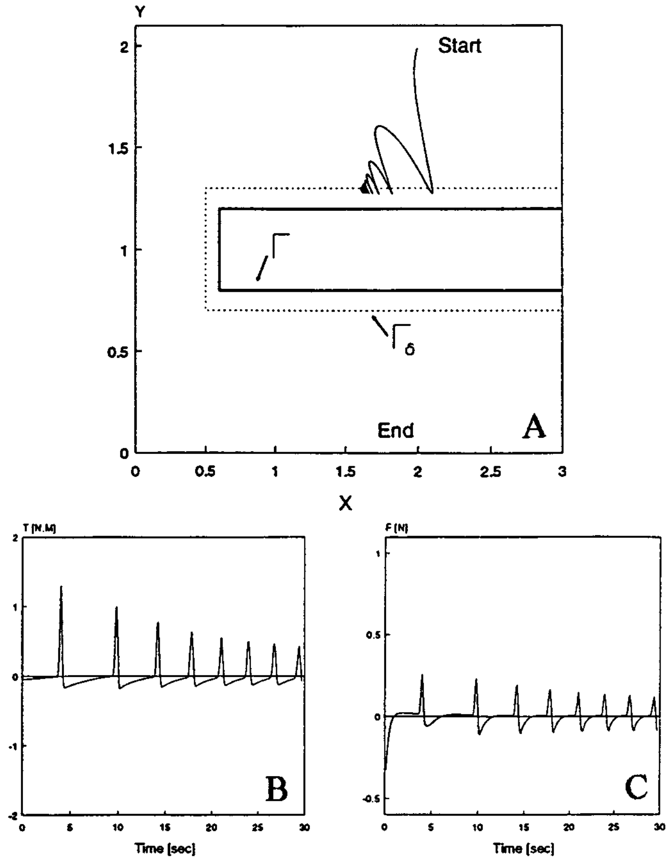


Fig. 13. Obstacle present, only PPC is used, (a) trajectory, (b) torque, (c) force.

PPC unit vector on Γ_{v+} and pointing in the positive direction of $\dot{q}$, $\mathrm{e}_n+(\dot{q}-)$, $\mathrm{e}_n-(\dot{q}+)$, and $\mathrm{e}_n-(\dot{q}-)$ are defined in a similar manner. In the following all possible combinations of the PPC's in q and $\dot{q}$ are examined to determine the situations of conflict:

1) $\mathrm{e}_n(q+)$ and $\mathrm{e}_n+(\dot{q}+)$ (Fig. 5.1)
This situation cannot occur since motion toward Γ implies that $q(t) < q(t-dt)$. This forces $\dot{q}$ to be negative. In other words, the PPC's in $\dot{q}$ and q can never be simultaneously active $(\alpha_1 \cdot \alpha_2 = 0)$. Such a situation is disregarded as a do not care situation.
2) $\mathrm{e}_n(q+)$ and $\mathrm{e}_n+(\dot{q}-)$ (Fig. 5.2)
This situation is similar to the one above (a do not care situation).
3) $\mathrm{e}_n(q+)$ and $\mathrm{e}_n-(\dot{q}+)$ (Fig. 5.3)
For this case, it is possible for q to be at Γ and $\dot{q}$ to be at Γ_v- at the same time. Here, $\mathrm{e}_n(q+)$ attempts to drive q in the positive direction making $q(t) > q(t-dt)$. In other words, $\mathrm{e}_n(q+)$ acts to drive $\dot{q}$ in the positive direction, which is in accord with what $\mathrm{e}_n-(\dot{q}+)$ tries to do. Therefore, no conflict can happen, an admissible situation.
4) $\mathrm{e}_n(q+)$ and $\mathrm{e}_n-(\dot{q}-)$ (Fig. 5.4)
In this case, while $\mathrm{e}_n(q+)$ attempts to drive $\dot{q}$ in the positive direction, $\mathrm{e}_n-(\dot{q}-)$ acts to drive $\dot{q}$ in the negative direction. This is a conflict situation that cannot be simultaneously enforced by $\mathrm{e}_n-(q+)$ and $\mathrm{e}_n-(\dot{q}-)$. Using similar arguments, it can be shown that:
5) $\mathrm{e}_n(q-)$ and $\mathrm{e}_n+(\dot{q}+)$ is a conflict situation;
6) $\mathrm{e}_n(q-)$ and $\mathrm{e}_n+(\dot{q}-)$ is an admissible situation;
7) $\mathrm{e}_n(q-)$ and $\mathrm{e}_n-(\dot{q}+)$ is a do not care situation; and
8) $\mathrm{e}_n(q-)$ and $\mathrm{e}_n-(\dot{q}-)$ is, also, a do not care situation.

For convenience, let us separately list the admissible situations

$$\text{a. } \mathrm{e}_n(q+) \quad \text{and} \quad \mathrm{e}_n-(\dot{q}+) \quad \text{b. } \mathrm{e}_n(q-) \quad \text{and} \quad \mathrm{e}_n+(\dot{q}-). \tag{46}$$

As can be seen, regardless of the direction of $\mathrm{e}_n(q)$, no conflict can arise as long as $\mathrm{e}_n(\dot{q})$ is pointing in a direction that attempts to reduce speed (i.e., $\mathrm{e}_n(\dot{q})$ pointing toward the origin of $\dot{q}$). Such a condition can be separately applied to the individual components of the N-dimensional $\mathrm{e}_n(\dot{\mathbf{q}})$.

Therefore, to guarantee that no conflict will arise, the following conditions have to be enforced for all $\mathbf{q} \in \Gamma_v$

$$\dot{q}_i e_{ni}(\dot{q}) \le 0 \quad i = 1, \ldots, N \tag{47}$$

where $\dot{q}_i$ and e_{ni} are the i'th component of $\dot{\mathbf{q}}$ and $\mathrm{e}_n(\dot{q})$, respectively.

Proof of Avoidance:

Proposition-3: For the control law

$$\mathbf{u}\Gamma_{ln} = \alpha_{\mathbf{1}}(\mathbf{q}, \dot{\mathbf{q}})\mathrm{e}_n(\mathbf{q}) + \alpha_{\mathbf{2}}(\mathbf{q}, \dot{\mathbf{q}})\mathrm{e}_n(\dot{\mathbf{q}}) \tag{48}$$

there exist an $\alpha_{\mathbf{1}}$ and an $\alpha_{\mathbf{2}}$ such that for the system in (14)

$$\mathbf{q} \notin O \quad \text{and} \quad \dot{\mathbf{q}} \notin O_v \quad \text{for all } t$$

provided that the conditions in (46) are satisfied.

Proof: Let d be a distance that is defied as

$$\begin{aligned} d &= \mathbf{1/2}(\mathbf{x}^t \mathrm{e}_n(x))^{\mathbf{2}} \\ &= \mathbf{1/2} x_n^{\mathbf{2}} \end{aligned} \tag{49}$$

where $\mathbf{x}^t = [\mathbf{q}^t \ \ \dot{\mathbf{q}}^t]$, and $\mathrm{e}_n^t(\mathbf{x}) = [\mathrm{e}_n^t(\mathbf{q}) \ \ \mathrm{e}_n^t(\dot{\mathbf{q}})]$. The time derivative of d is

$$\dot{d} = x_n \cdot \dot{x}_n = (\mathbf{x}^t \mathrm{e}_n(\mathbf{x}))(\dot{\mathbf{x}}^t \mathrm{e}_n(\mathbf{x})). \tag{50}$$

For avoidance to be successful, $\dot{d}$ must be

$$\dot{d} \ge 0.$$

Since the state is assumed to be initially outside the avoidance regions (i.e., $x_n(0) > 0$), the conditions for successful avoidance reduces to

$$\dot{\mathbf{x}}^t \mathrm{e}_n(\mathbf{x}) \ge 0. \tag{51}$$

The above expression is equivalent to

$$\begin{aligned} \dot{\mathbf{q}}^t \mathrm{e}_n(\mathbf{q}) + \ddot{\mathbf{q}} \mathrm{e}_n(\dot{\mathbf{q}}) = \mathrm{e}_n(\mathbf{q})^t \int_{t\mathbf{0}}^{t} [\mathbf{f}(\mathbf{q}, \dot{\mathbf{q}}) + \mathbf{D}^{-1}(\mathbf{q})\mathbf{u}_l] \, dt \\ + \mathrm{e}_n(\dot{\mathbf{q}})^t [\mathbf{f}(\mathbf{q}, \dot{\mathbf{q}}) + \mathbf{D}^{-1}(\mathbf{q})\mathbf{u}_l] \ge 0. \end{aligned} \tag{52}$$

Substituting $\mathbf{u}\Gamma_{ln}$ in (52), we have

$$\begin{aligned} \mathrm{e}_n^t(\mathbf{q}) \int_{t\mathbf{0}}^{t} [\mathbf{f}(\mathbf{q}, \dot{\mathbf{q}}) + \alpha_{\mathbf{1}}(\mathbf{q}, \dot{\mathbf{q}})\mathbf{D}^{\mathbf{-1}}(\mathbf{q})\mathrm{e}_n(\mathbf{q})] \, dt \\ + \mathrm{e}_n^t(\dot{\mathbf{q}})[\mathbf{f}(\mathbf{q}, \dot{\mathbf{q}}) + \alpha_{\mathbf{2}}(\mathbf{q}, \dot{\mathbf{q}})\mathbf{D}^{\mathbf{-1}}(\mathbf{q})\mathrm{e}_n(\dot{\mathbf{q}})] \end{aligned}$$

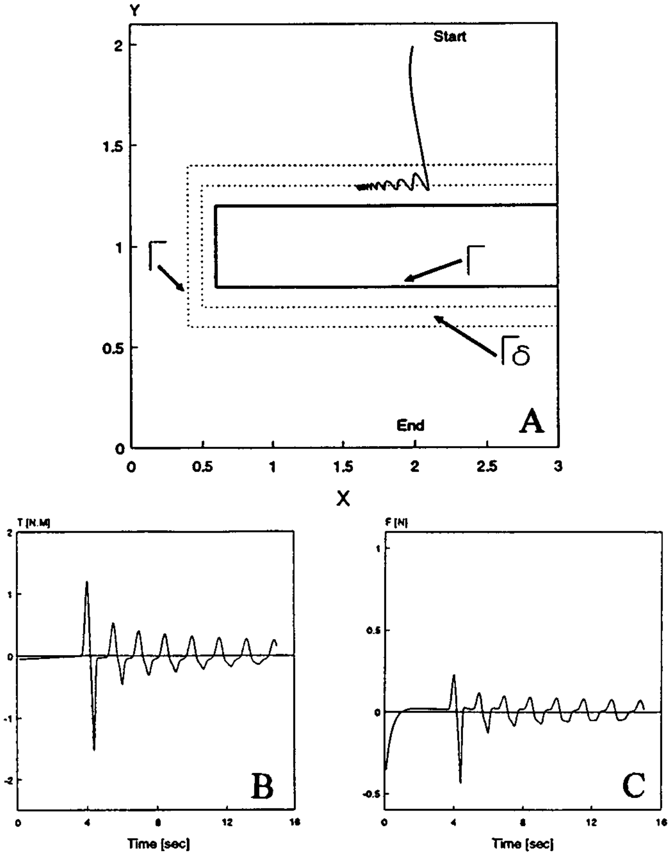


Fig. 14. Obstacle present, PPC, and clamping control used: (a) trajectory, (b) torque, and (c) force.

$$+\int_{t0}^{t}[\alpha_{\mathbf{1}}(\mathbf{q},\dot{\mathbf{q}})\mathrm{e}_n^t(\dot{\mathbf{q}})\mathbf{D}^{-1}(\mathbf{q})\mathrm{e}_n(\mathbf{q})]\,dt + \left[\alpha_{\mathbf{2}}(\mathbf{q},\dot{\mathbf{q}})\mathrm{e}_n^t(\mathbf{q})\mathbf{D}^{-1}(\mathbf{q})\mathrm{e}_n(\dot{\mathbf{q}})\right]. \tag{53}$$

The first two terms of the above expression are guaranteed to be greater than or equal to zero by enforcing conditions (31), (34), and (42). This reduces the avoidance conditions in the joint $\mathbf{q}-\dot{\mathbf{q}}$ space to guaranteeing that

$$\int_{t0}^{t}\left[\alpha_{\mathbf{1}}(\mathbf{q},\dot{\mathbf{q}})\mathrm{e}_n^t(\dot{\mathbf{q}})\mathbf{D}^{-1}(\mathbf{q})\mathrm{e}_n(\mathbf{q})\right]dt + \left[\alpha_{\mathbf{2}}(\mathbf{q},\dot{\mathbf{q}})\mathrm{e}_n^t(\mathbf{q})\mathbf{D}^{-1}(\mathbf{q})\mathrm{e}_n(\dot{\mathbf{q}})\right] \geq 0. \tag{54}$$

Before tackling the above expression, the relation between a unit vector in the position space and the corresponding one in the velocity space needs to be examined. Assume that the robot is moving along a trajectory (q_n) that corresponds to the unit vector $\mathrm{e}_n(\mathbf{q})$ in the position space. The corresponding speed can be calculated as

$$\begin{aligned}\dot{\mathbf{q}}_n &= \frac{q_n(t)-q_n(t-dt)}{dt}\mathrm{e}_n(\mathbf{q})\\ &= \frac{\mathbf{q}(t)^t\mathrm{e}_n(\mathbf{q})-\mathbf{q}(t-dt)^t\mathrm{e}_n(\mathbf{q})}{dt}\mathrm{e}_n(\mathbf{q}).\end{aligned} \tag{55}$$

The velocity unit vector $\mathrm{e}_n(\dot{\mathbf{q}})$ is computed as

$$\begin{aligned}\mathrm{e}_n(\dot{\mathbf{q}}) &= \frac{\dot{\mathbf{q}}_n}{|\dot{\mathbf{q}}_n|}\\ &= \frac{(\mathbf{q}(t)^t\mathrm{e}_n(\mathbf{q}))\mathrm{e}_n(\mathbf{q})-(\mathbf{q}(t-dt)^t\mathrm{e}_n(\mathbf{q}))\mathrm{e}_n(\mathbf{q})}{|\mathbf{q}(t)^t\mathrm{e}_n(\mathbf{q})-\mathbf{q}(t-dt)^t\mathrm{e}_n(\mathbf{q})|}\\ &= \frac{\mathbf{q}(t)^t\mathrm{e}_n(\mathbf{q})-\mathbf{q}(t-dt)^t\mathrm{e}_n(\mathbf{q})}{|\mathbf{q}(t)^t\mathrm{e}_n(\mathbf{q})-\mathbf{q}(t-dt)^t\mathrm{e}_n(\mathbf{q})|}\cdot\mathrm{e}_n(\mathbf{q}).\end{aligned} \tag{56}$$

In other words, $\mathrm{e}_n(\dot{\mathbf{q}}) = \mp\mathrm{e}_n(\mathbf{q})$.

From (46) it can be seen that for the constraints on the avoidance regions to be admissible, both $\mathrm{e}_n(\mathbf{q})$ and $\mathrm{e}_n(\dot{\mathbf{q}})$ must have the same sign (i.e., at the intersection of O and O_v $\mathrm{e}_n(\mathbf{q}) = \mathrm{e}_n(\dot{\mathbf{q}})$). Therefore, (54) reduces to

$$\int_{t^0}^{t}\left[\alpha_{\mathbf{1}}(\mathbf{q},\dot{\mathbf{q}})\mathrm{e}_n^t(\mathbf{q})\mathbf{D}^{-1}(\mathbf{q})\mathrm{e}_n(\mathbf{q})\right]dt + \left[\alpha_{\mathbf{2}}(\mathbf{q},\dot{\mathbf{q}})\mathrm{e}_n^t(\dot{\mathbf{q}})\mathbf{D}^{-1}(\mathbf{q})\mathrm{e}_n(\dot{\mathbf{q}})\right]. \tag{57}$$

Since $\alpha_{\mathbf{1}}$ and $\alpha_{\mathbf{2}}$ are restricted to be positive, and $\mathbf{D}^{-1}$ is a positive definite matrix, the above expression is always greater than or equal to zero. Therefore, $\dot{d}$ is always greater than or equal to zero and avoidance in the joint position-velocity space is guaranteed by simply guaranteeing avoidance in the separate spaces along with the consistency of the avoidance constraints. □

D. The BLAC in Position Space

The BLAC begins to function once the state is about to touch Γ (The PPC will prevent the state from actually touching Γ.) The motion of the robot is then restricted to the surface of the obstacle (Γ) and driven to a location on Γ where $\mathbf{u}_g$ regains the ability to steer motion to the target along an obstacle-free path. Finally, command of the robot is transferred to $\mathbf{u}_g$ once that location is reached. The first step in implementing the above behavior is to partition Γ into two parts, Γ_r and Γ_o (PPC is assumed to be present), so that at $t = t_{\mathbf{1}}$

$$\begin{array}{llll}\text{if} & \mathbf{q}(t_{\mathbf{1}})\in\Gamma_r & \text{then} & \lim\limits_{t\to\infty}\mathbf{q}(t)\to\mathbf{q}_r\\ \text{and if} & \mathbf{q}(t_{\mathbf{1}})\in\Gamma_o & \text{then} & \lim\limits_{t\to\infty}\mathbf{q}(t)\in\Gamma_o.\end{array} \tag{58}$$

The second step is to clamp $\mathbf{q}$ to the Γ_o part of Γ. The final step is to construct a control field that is tangent to Γ and has the ability to drive motion toward $\Gamma_r(\mathbf{u}_{\Gamma_{lt}})$. To construct such a field, a polar coordinate system is embedded in Γ. This system spans only one degree of freedom of the $(N-1)$ degrees of freedom that are available to the unit vector field $\mathrm{e}_t(\mathbf{q})$ which is tangent to Γ. In this coordinate system a point on Γ is described by the vector ξ

$$\xi = [\xi_{\mathbf{1}}\ldots\xi_{N-\mathbf{1}}]^t, \quad \xi\in Q \tag{59}$$

where Q is the domain on which ξ is defined ($Q\in R^{N-\mathbf{1}}$, Q is an obstacle-free space). The polar coordinates in Q are simple star-shaped lines that sink in the focal point ξ_r. Any point in Q (and in turn on Γ) can be uniquely determined by specifying its distance from ξ_r $(|\xi|)$ and a set of angles $(\theta(\xi))$ that are measured from a reference line in Q. Here $\mathbf{q}$ is a mapping between Q and Γ $(\mathbf{q}\colon Q\to\Gamma)$. This mapping (see Appendix I) is one-to-one and onto. The focal point of the coordinate system (ξ_r) is chosen such that $\mathbf{q}(\xi_r)\in\Gamma_r$. Ensuring global convergence of ξ to ξ_r (i.e., $\mathbf{q}(\xi)\to\mathbf{q}(\xi_r)$), ensures that the robot will enter Γ_r, after which $\mathbf{u}_g$ drives it to $\mathbf{q}_r$.

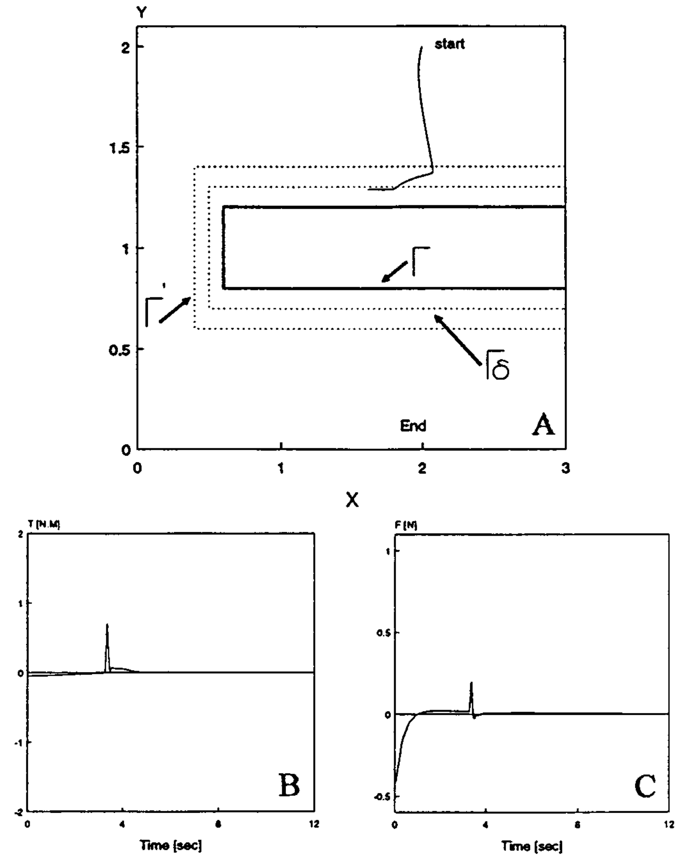


Fig. 15. Obstacle present, PPC, clamping, and nonlinear anisotropic damping controls used: (a) trajectory, (b) torque, and (c) force.

Proposition-4: For a control of the form

$$\mathbf{u}_{\Gamma_{lt}}(\xi) = -\beta_{\mathbf{1}}(\mathbf{q}(\xi)) \cdot \frac{\xi - \xi_r}{\|\xi - \xi_r\|} \quad \mathbf{q}(\xi) \in \Gamma \tag{60}$$

where $\beta_{\mathbf{1}}$ is a scalar function, there exists a $\beta_{\mathbf{1}} > 0$ such that

$$\operatorname*{Lim}_{t \to \infty} \xi \to \xi_r.$$

Proof: To prove global asymptotic convergence of ξ to ξ_r, it is shown that the time derivative of the following Liapunov function

$$\Xi(\xi) = \frac{\mathbf{1}}{\mathbf{2}}(\xi - \xi_r)^t(\xi - \xi_r) \tag{61}$$

is always negative definite

$$\dot{\Xi}(\xi) < 0. \tag{62}$$

To begin proving the above, it must first be noticed that as a consequence of the passivity property of robotics manipulators [64], the system forces that determine convergence to a point in the position space of the manipulator's state space have the form [63]

$$\mathbf{f}_q(\mathbf{q}) = \lim_{\dot{\mathbf{q}} \to \mathbf{0}} \mathbf{f}(\mathbf{q}, \dot{\mathbf{q}}). \tag{63}$$

Consequently, the system equation that governs the motion of ξ is

$$\dot{\xi} = \mathbf{f}_{q_t}(\mathbf{q}(\xi)) + \mathbf{D}^{-\mathbf{1}}(\mathbf{q}(\xi))\mathbf{u}_{\Gamma_{lt}}(\xi) \tag{64}$$

where $\mathbf{f}_{q_t}(\mathbf{q}(\xi), \dot{\mathbf{q}}) = \mathbf{e}_t^t(\mathbf{q}(\xi))\mathbf{f}_q(\mathbf{q}(\xi))$. The time derivative of the above Liapunov function is

$$\begin{aligned}\dot{\Xi} &= (\xi - \xi_r)^t\dot{\xi} \\ &= (\xi - \xi_r)^t(\mathbf{f}_{q_t}(\mathbf{q}(\xi)) + \mathbf{D}^{-\mathbf{1}}(\mathbf{q}(\xi))\mathbf{u}\Gamma_{lt}(\xi)) \\ &= -\beta_{\mathbf{1}}(\mathbf{q}(\xi))\frac{(\xi - \xi_r)^t\mathbf{D}^{-\mathbf{1}}(\mathbf{q}(\xi))(\xi - \xi_r)}{\|\xi - \xi_r\|} \\ &\quad + (\xi - \xi_r)^t\mathbf{f}_{q_t}(\mathbf{q}(\xi)).\end{aligned} \tag{65}$$

Let $\mathbf{e}_\rho(\xi)$ be the radial basis vector in Q

$$\mathbf{e}_\rho(\xi) = \frac{\xi - \xi_r}{\|\xi - \xi_r\|} \tag{66}$$

and $\mathbf{e}_\tau(\xi)$ be the basis vector in Q that is normal to $\mathbf{e}_\rho$. Let $\mathbf{f}_{q_t}$ be represented in terms of these two basis vectors

$$\mathbf{f}_{q_t} = \eta_{\mathbf{1}}(\mathbf{q}(\xi))\mathbf{e}_\rho(\xi) + \eta_{\mathbf{2}}(\mathbf{q}(\xi))\mathbf{e}_\tau(\xi) \tag{67}$$

where $\eta_{\mathbf{1}} = \mathbf{f}q_t^t\mathbf{e}_\rho(\xi)$, and $\eta_{\mathbf{2}} = \mathbf{f}q_t^t\mathbf{e}_\tau(\xi)$. Substituting the above term in $\dot{\Xi}$ we get

$$\begin{aligned}\dot{\Xi} &= -\beta_{\mathbf{1}}(\mathbf{q}(\xi))\frac{(\xi - \xi_r)^t\mathbf{D}^{-\mathbf{1}}(\mathbf{q}(\xi))(\xi - \xi_r)}{\|\xi - \xi_r\|} \\ &\quad + \|\xi - \xi_r\|\mathbf{e}_\rho^t(\xi)(\eta_{\mathbf{1}}(\mathbf{q}(\xi))\mathbf{e}_\rho(\xi) \\ &\quad + \eta_{\mathbf{2}}(\mathbf{q}(\xi))\mathbf{e}_\tau(\xi)) \\ &= -\beta_{\mathbf{1}}(\mathbf{q}(\xi))\frac{(\xi - \xi_r)^t\mathbf{D}^{-\mathbf{1}}(\mathbf{q}(\xi))(\xi - \xi_r)}{\|\xi - \xi_r\|} \\ &\quad + \eta_{\mathbf{1}}(\mathbf{q}(\xi)) \cdot \|\xi - \xi_r\|.\end{aligned} \tag{68}$$

To guarantee that the above is negative definite, the following inequality must hold

$$\begin{aligned}&\beta_{\mathbf{1}}(\mathbf{q}(\xi))\frac{(\xi - \xi_r)^t\mathbf{D}^{-\mathbf{1}}(\mathbf{q}(\xi))(\xi - \xi_r)}{\|\xi - \xi_r\|} \\ &\quad > \eta_{\mathbf{1}}(\mathbf{q}(\xi)) \cdot \|\xi - \xi_r\|.\end{aligned} \tag{69}$$

A choice for $\beta_{\mathbf{1}}$ which guarantees that the above inequality hold is

$$\beta_{\mathbf{1}}(q(\xi)) > \frac{\eta_{\mathbf{1}}(\mathbf{q}(\xi)) \cdot \|\xi - \xi_r\|^{\mathbf{2}}}{(\xi - \xi_r)^t\mathbf{D}^{-\mathbf{1}}(\mathbf{q}(\xi))(\xi - \xi_r)}. \tag{70}$$

□

E. Computing the Exit Point $(\mathbf{q}(\xi_r))$

While it may be desirable to partition Γ into $\Gamma_o \cup \Gamma_r$, it is nevertheless sufficient to compute only one point $\mathbf{q}(\xi_r)$ on Γ_r in order to construct the BLAC. The following steps are recommended for computing $\mathbf{q}(\xi_r)$ for a general nonconvex region and a globally, asymptotically-convergent global nonlinear control field $\mathbf{u}_g$ (Fig. 6)

1) Choose a point $\mathbf{q}_s$ on Γ.
2) Construct the following differential equation

$$\dot{\mathbf{q}} = \mathbf{f}_q(\mathbf{q}). \tag{71}$$

3) With the boundary steering control disabled $(\mathbf{u}_{\Gamma_l} = 0)$, forward traverse the flow lines of $\mathbf{f}_q$ toward the target using (71). Motion should start from $\mathbf{q}_s(\mathbf{q}(0) = \mathbf{q}_s)$ and end at $\mathbf{q}_\varepsilon$, where $|\mathbf{q}_r - \mathbf{q}_\varepsilon| = \varepsilon$, and $\varepsilon \to 0$.

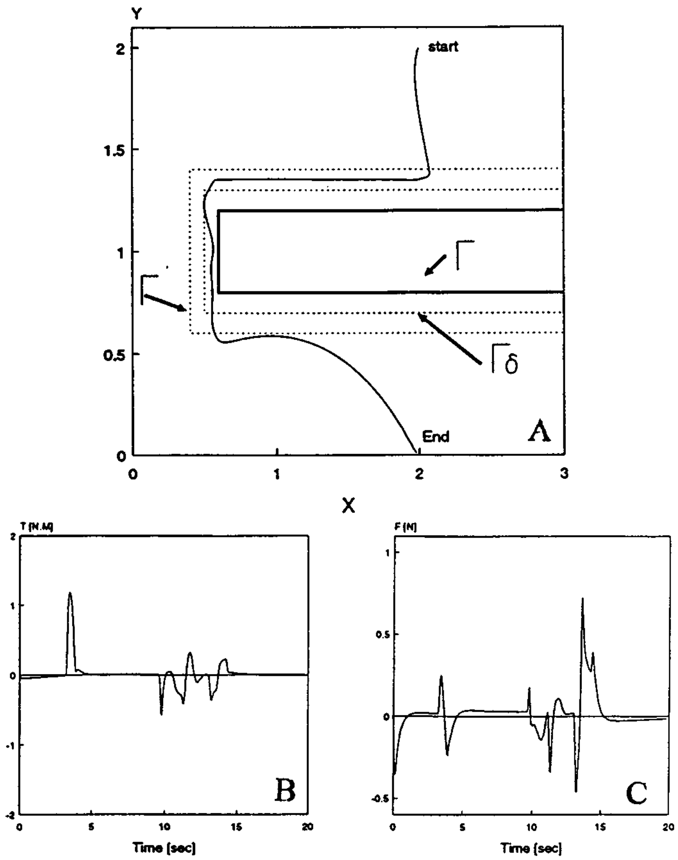


Fig. 16. Obstacle present, PPC, LAC, clamping, nonlinear anisotropic damping controls used: (a) trajectory, (b) torque, and (c) force.

4) Now, starting from $\mathbf{q}_\varepsilon$, traverse the field lines of $\mathbf{f}_q$ backward toward the obstacle using the equation

$$\dot{\mathbf{q}} = -\mathbf{f}_q(\mathbf{q}). \tag{72}$$

5) The first point that the backward path touches on Γ is the desired point $\mathbf{q}(\xi_r)$.

Although it is not necessary for computing the BLAC, Γ_r can be fully computed by repeating the above procedure for a sufficiently dense set of points (excluding the $\mathbf{q}_s$ and $\mathbf{q}_\varepsilon$ points from previous trials) that are used as starting points for (71). The remaining part of Γ is taken as Γ_o.

V. The Steering Control

In the previous section the BPPC and BLAC are constructed in the local coordinates of the obstacle. This is carried out under the assumption that a normal and tangential set of coordinates already exist. In this section a procedure is suggested for constructing these coordinates in a manner that enables their direct utilization for motion steering in the natural coordinates of the robot. These coordinates, along with the BPPC and BLAC, are used for constructing smooth PPC and LAC components that would gradually decelerate the robot, prevent collision, and deflect motion toward $\mathbf{q}(\xi_r)$ where it is subsequently steered by $\mathbf{u}_g$ to $\mathbf{q}_r$. These components occupy a finite region $(O\delta)$ that surrounds O. The suggested procedure constructs two scalar harmonic potential fields ($V_{\mathbf{1}n}(\mathbf{q})$ and $V_{\mathbf{1}t}(\mathbf{q})$) one for each component of the steering control. The potential fields are constructed so that the resulting configuration of the gradient flow-lines on Γ matches that of the obstacle's local coordinates (i.e.,

$$\frac{\nabla V_{\mathbf{1}n}(\mathbf{q})}{|\nabla V_{\mathbf{1}n}(\mathbf{q})|} = \mathbf{e}_n(\mathbf{q}) \quad \mathbf{q} \in \Gamma$$

and

$$\frac{\nabla V_{\mathbf{1}t}(\mathbf{q})}{|\nabla V_{\mathbf{1}t}(\mathbf{q})|} = \mathbf{e}_t(\mathbf{q})). \tag{73}$$

It ought to be noticed that $V_{\mathbf{1}t}$ can be used to construct an invertible mapping between ξ and $\mathbf{q} \in \Gamma$ (see Appendix I). The PPC and LAC are each divided into two components: a vector phase field component, and a scalar magnitude field component

$$\begin{aligned}\mathbf{u}_{ln}(\mathbf{q}, \dot{\mathbf{q}}) &= M_n(\mathbf{q}, \dot{\mathbf{q}}) \cdot \mathbf{Q}_n(\mathbf{q})\\ \mathbf{u}_{lt}(\mathbf{q}) &= M_t(\mathbf{q}) \cdot Q_t(\mathbf{q})\end{aligned} \tag{74}$$

where $\mathbf{Q}_n$ and $\mathbf{Q}_t$ are the basis vector phase fields for the normal and tangential coordinates respectively, M_n and M_t are the scalar magnitude fields for the normal and tangent coordinates, respectively.

A. The PPC

To generate $\mathbf{Q}_n$, the following SBVP is solved [Fig. 7(a)]

$$\nabla^2 V_{\mathbf{1}n}(\mathbf{q}) = 0 \tag{75}$$

subject to

$$V_{\mathbf{1}n}(\mathbf{q})|_\Gamma = C, \quad \text{and} \quad V_{\mathbf{1}n}(\mathbf{q})|_{\Gamma\delta} = 0 \quad C > 0$$
$$\mathbf{Q}_n(\mathbf{q}) = \frac{\nabla V_{\mathbf{1}n}(\mathbf{q})}{\|\nabla V_{\mathbf{1}n}(\mathbf{q})\|}.$$

The magnitude field is generated by solving the SBVP

$$\nabla^2 V_{\mathbf{2}n}(\mathbf{q}) = 0 \tag{76}$$

subject to

$$V_{\mathbf{2}n}(\mathbf{q})|_{\Gamma\delta_d} = 1 \quad \text{and} \quad V_{\mathbf{2}n}(\mathbf{q})||_{\Gamma\delta} = 0$$

$$M_n(\mathbf{q}, \dot{\mathbf{q}}) = \begin{bmatrix} \alpha_{\mathbf{1}}(\mathbf{q}, \dot{\mathbf{q}}) & \mathbf{q} \in O\delta_d \\ \alpha_{\mathbf{1}}(\mathbf{q}', \dot{\mathbf{q}})V_{\mathbf{2}n}(\mathbf{q}) & \mathbf{q} \in O\delta, \quad \mathbf{q}' \in \Gamma\delta_d \end{bmatrix}$$

If the PPC is to clamp the robot to Γ_o as well, the following additional boundary condition is needed:

$$V_{\mathbf{2}n}|_{\Gamma_o'} = -1 \tag{77}$$

where Γ_o' is the portion of Γ' that corresponds to Γ_o, and Γ' is an equipotential surface of $V_{\mathbf{1}n}$ inside $O\delta$ chosen equal to $C/2$.

B. The LAC

The following steps are used to construct the LAC component [Fig. 7(b)]:

1) Choose $\mathbf{q}(\xi_r)$ inside Γ_r, and $\mathbf{q}(\xi_n)$ inside Γ_o.
2) Construct the following lines

$$\rho_r = \{\mathbf{q}\colon \dot{\mathbf{q}}(t) = -\mathbf{Q}_n(\mathbf{q}), 0 \le t \le \tau, \mathbf{q}(0) = \mathbf{q}(\xi_r)$$
$$\mathbf{q}(\tau) \in \Gamma\delta\}$$

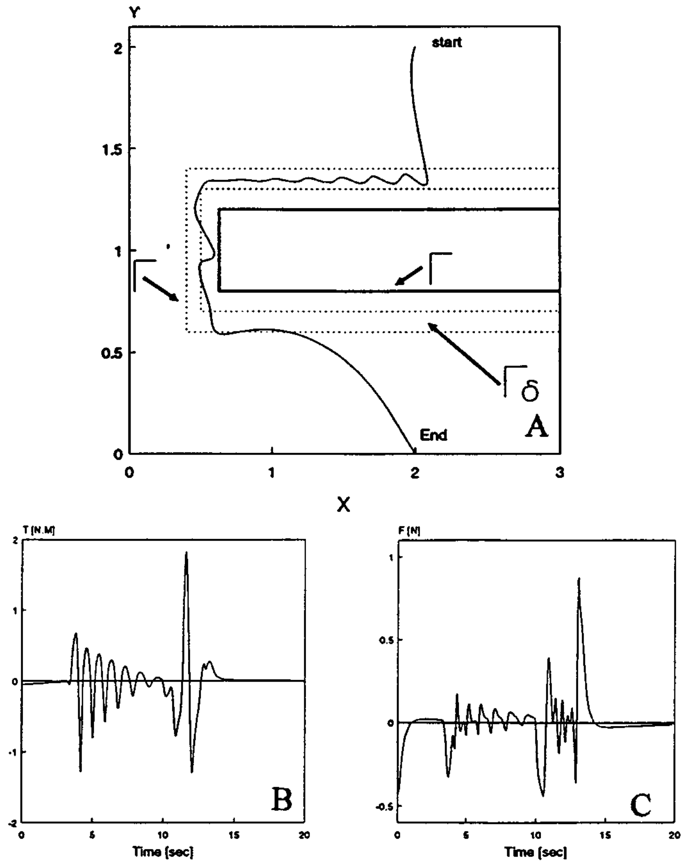


Fig. 17. Same as Fig. 16 without nonlinear anisotropic damping control, (a) trajectory, (b) torque, and (c) force.

$$\rho_n = \{\mathbf{q}: \dot{\mathbf{q}}(t) = -\mathbf{Q}_n(\mathbf{q}), 0 \le t \le \tau, \mathbf{q}(0) = \mathbf{q}(\xi_n), \mathbf{q}(\tau) \in \Gamma\delta\}. \tag{78}$$

3) Solve the following BVP

$$\nabla^2 V_{\mathbf{1}t}(\mathbf{q}) = 0 \tag{79}$$

subject to

$$V_{\mathbf{1}t}(\mathbf{q})|_{\rho_r} = 0, \quad \text{and} \quad V_{\mathbf{1}t}(\mathbf{q})|_{\rho_n} = C \quad C > 0$$
$$\partial V_{\mathbf{1}t}(\mathbf{q})/\partial_n = 0 \quad \text{at } \Gamma', \Gamma, \quad \text{and} \quad \Gamma\delta$$
$$\mathbf{Q}_t(\mathbf{q}) = \frac{\nabla V_{\mathbf{1}t}(\mathbf{q})}{\|\nabla V_{\mathbf{1}t}(\mathbf{q})\|}.$$

4) Compute the magnitude field by solving the following BVP:

$$\nabla^2 V_{\mathbf{2}t}(\mathbf{q}) = 0 \tag{80}$$

subject to

$$V_{\mathbf{2}t}(\mathbf{q}) = \beta_{\mathbf{1}}(\mathbf{q})|_{\Gamma'}, \quad V_{\mathbf{2}t}(\mathbf{q}) = \beta_{\mathbf{1}}(\mathbf{q})|_{\Gamma}, \quad \text{and} \quad V_{\mathbf{2}t}(\mathbf{q})|_{\Gamma\delta} = 0$$

$$M_t(\mathbf{q}) = V_{\mathbf{2}t}(\mathbf{q}).$$

Existence and uniqueness of the solution of the above BVP were proven in [62]. It ought to be mentioned that ρ_r and ρ_n are both specified to give the designer more control over the field. It is enough to specify ρ_r (a south pole) alone in the above generating BVP in order for a north pole (ρ_n) to automatically form in the resulting field. Since the distance from a north pole to a south pole is the same regardless of the direction from which motion proceeds, a steering control constructed in this way will sweep the robot along the shortest path around the obstacle to Γ_r.

C. *Orthogonality of* $\mathbf{Q}_n$ *to* $\mathbf{Q}_t$

Here a proof of the orthogonality of the PPC to the LAC is supplied.

Proposition-5: The phase field that is constructed in V.A $(\mathbf{Q}_n(\mathbf{q}))$ is orthogonal to the one constructed in V.B $(\mathbf{Q}_t(\mathbf{q}))$

$$\nabla V_{\mathbf{1}n}^t(\mathbf{q}) \nabla V_{\mathbf{1}t}(\mathbf{q}) \equiv 0. \tag{81}$$

Proof: Since both the BVP's that generate $V_{\mathbf{1}n}$ and $V_{\mathbf{1}t}$ have unique solutions, the flow lines that are marked by $\nabla V_{\mathbf{1}n}$ do not intersect each other. The same goes for the flow lines that are marked by $\nabla V_{\mathbf{1}t}$. Consequently, the equipotential contours that are associated with any of the gradient-flow are parallel and do not intersect. With this in mind, it is easy to see that proving the parallelism of the gradient flow from one potential field to the family of equipotential contours of the other is equivalent to proving that the flow lines of the potential flows orthogonally intersect each other.

Let ρ_n be the flow line of $\nabla V_{\mathbf{1}n}$ $(\mathbf{Q}_n)$ which is defined in V.A. Note that by choice of boundary conditions, ρ_n is also an equipotential line of $V_{\mathbf{1}t}$. Let ρ'_ε be another flow line of $\nabla V_{\mathbf{1}n}$ that starts from Γ a small distance ε $(\varepsilon \to 0)$ away from ρ_n (Fig. 8). Let ρ_ε be an equipotential line of $V_{\mathbf{1}t}$ that also starts from ε. Since ρ_n is simultaneously an equipotential line of $V_{\mathbf{1}t}$ and a gradient flow line of $V_{\mathbf{1}n}$, ρ_n is parallel to both ρ'_ε and ρ_ε. From the uniqueness of the solutions of both BVP's, the initial position ε defines one and only one gradient flow line of $V_{\mathbf{1}n}$ and equipotential line of $V_{\mathbf{1}t}$. Therefore, ρ'_ε and ρ_ε must be identical. By repeatedly applying this argument to consecutive gradient flow and equipotential lines of $V_{\mathbf{1}n}$ and $V_{\mathbf{1}t}$ respectively, it can be shown that the gradient flow of $V_{\mathbf{1}n}$ is identical to the equipotential lines of $V_{\mathbf{1}t}$. Since equipotential lines intersect their gradient flow lines orthogonally, $\nabla V_{\mathbf{1}n}$ lines intersect $\nabla V_{\mathbf{1}t}$ lines orthogonally; hence,

$$\nabla V_{\mathbf{1}n}^t \nabla V_{\mathbf{1}t} = 0. \tag{82}$$

$\mathbf{Q}_n$ and $\mathbf{Q}_t$ may be viewed as boundary-fitted, general, curvilinear coordinates that are used for synthesizing the control inside the admissible region of state space (i.e., workspace). Other methods for building coordinate systems may be found in [77] and [78].

D. *Implementation*

It is possible to obtain a closed form solution to the Laplace Equation for simple or even relatively involved cases [65]. However, one should take into consideration that the path-planning stage is an intermediate module in a robotics system. This stage takes information from the sensors and the system operator (numerical data about the target and obstacle), and feeds processed

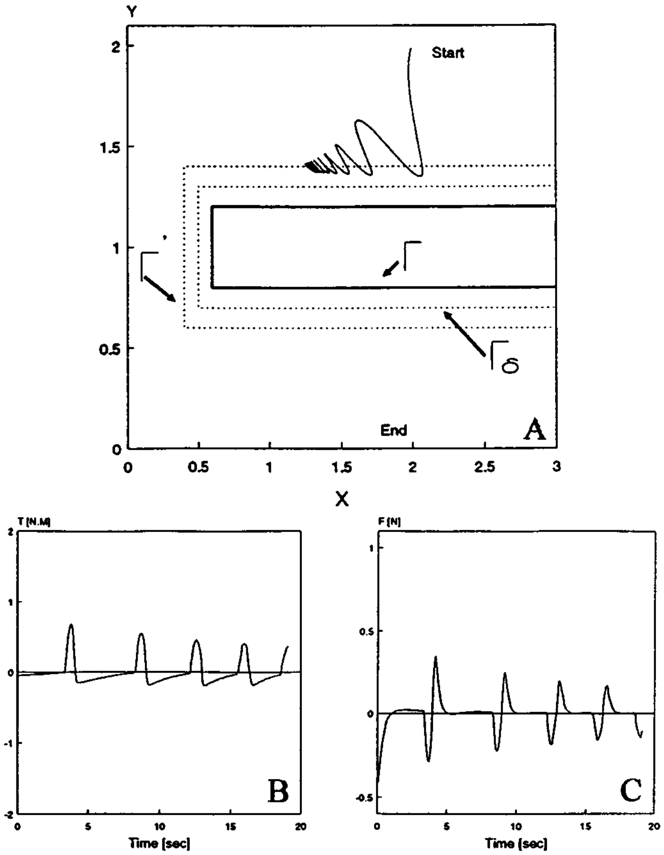


Fig. 18. Both damping and clamping controls removed, (a) trajectory, (b) torque, and (c) force.

information to the motion actuators. To suit the nature of such a task, numerical methods have to be used for the solution. There are different numerical techniques that can be used to solve PDE's [66]–[68]. It is important to choose a method that is compatible with the type of information describing the workspace.

One technique for solving a given BVP is called the Boundary Element Method (BEM). This technique approximates the solution to the field by discritizing

$$\begin{aligned} V(\mathbf{r}) &= \frac{1}{2\pi}\oint_\Gamma \left(\frac{\partial V(\mathbf{q})}{\partial n}\cdot G(\mathbf{r},\mathbf{q}) - V(\mathbf{q})\cdot\frac{\partial G(\mathbf{r},\mathbf{q})}{\partial \mathbf{n}}\right) d\Gamma \\ \frac{\partial V(\mathbf{r})}{\partial q_i(\mathbf{r})} &= \frac{1}{2\pi}\oint_\Gamma \left(\frac{\partial V(\mathbf{q})}{\partial \mathbf{n}}\cdot\frac{\partial G(\mathbf{r},\mathbf{q})}{\partial q_i(\mathbf{r})}\right. \\ &\qquad \left. - V(\mathbf{q})\cdot\frac{\partial}{\partial q_i(\mathbf{r})}\frac{\partial G(\mathbf{r},\mathbf{q})}{\partial \mathbf{n}}\right) d\Gamma \end{aligned} \tag{83}$$

where Γ is the closed surface surrounding $O\delta$, $\mathbf{r}$ is a point inside $O\delta$, $\mathbf{q}$ is a point on Γ, and $G(\mathbf{r},\mathbf{q})$ is the fundamental solution of the Laplace BVP (Green's function) in the specified dimension. A list of these functions can be found in [69]. Details on how to apply this method can be found in [70] and [71]. This technique has two properties that are instrumental to an efficient implementation. The first is its ability to reduce the dimensionality and, in turn, the complexity of the problem by one. The second has to do with generating the field from its value at the boundary. This is of a considerable importance since most of the methods describing the workspace represent it by encoding its boundary contours. It ought to be noted that all the inputs to (83) are specified in terms of the local coordinates of the obstacles. However, the generated output (steering control) is produced in the natural coordinates of the robot.

VI. Nonlinear Anisotropic Damping of Motion

A position PPC acts to prevent motion beyond a specified level of the NC flow contours that are made to coincide with the contours of the obstacles. A need may arise (see Example-2) where instead of strictly forcing motion away from certain sectors in the workspace, it is only required that motion be discouraged (damped) from proceeding along the directions (flowlines) that lead to these regions (Fig. 9). In the following, a damping control that can achieve the above task is suggested.

Proposition-6: A control of the form

$$\mathbf{u}_d(\mathbf{q},\dot{\mathbf{q}}) = -M(\mathbf{q})[\dot{\mathbf{q}}^t\mathbf{Q}(q)]\mathbf{Q}(\mathbf{q}) \tag{84}$$

can damp motion along the flow-lines of $\mathbf{Q}$, where $\mathbf{Q}(\mathbf{q})$ is the basis vector phase field that define the directions along which motion is impeded, and $M(\mathbf{q})$ is a positive scalar field that controls the degree of damping.

Proof: Let $\mathbf{q}_Q(\mathbf{q})$ be the component of $\mathbf{q}$ that is in phase with $\mathbf{Q}(\mathbf{q})$

$$\mathbf{q}_Q(\mathbf{q}) = \mathbf{q}^t\mathbf{Q}(\mathbf{q}). \tag{85}$$

Since $\mathbf{Q}(\mathbf{q})$ does not vary with time, we have

$$\dot{\mathbf{q}}_Q(\mathbf{q}) = \dot{\mathbf{q}}^t\mathbf{Q}(\mathbf{q})$$

and

$$\ddot{\mathbf{q}}_Q(\mathbf{q}) = \ddot{\mathbf{q}}^t\mathbf{Q}(\mathbf{q}). \tag{86}$$

The system equation, as seen from the $\mathbf{Q}$ coordinates, has the form

$$\begin{aligned} \ddot{\mathbf{q}}^t\mathbf{Q}(\mathbf{q}) &= \mathbf{Q}^t(\mathbf{q})\mathbf{f}(\mathbf{q},\dot{\mathbf{q}}) + \mathbf{Q}^t(\mathbf{q})\mathbf{D}^{-1}(\mathbf{q})\mathbf{u} \\ \ddot{\mathbf{q}}_Q(\mathbf{q}) &= \mathbf{f}_Q(\mathbf{q},\dot{\mathbf{q}}) + \mathbf{Q}^t(\mathbf{q})\mathbf{D}^{-1}(\mathbf{q})\mathbf{u} \end{aligned} \tag{87}$$

where $\mathbf{f}_Q(\mathbf{q},\dot{\mathbf{q}}) = \mathbf{Q}^t(q)\mathbf{f}(\mathbf{q},\dot{\mathbf{q}})$. Substituting $\mathbf{u}_d$ in the above equation

$$\begin{aligned} \ddot{\mathbf{q}}_Q(\mathbf{q}) &= \mathbf{f}_Q(\mathbf{q},\dot{\mathbf{q}}) + \mathbf{Q}^t(\mathbf{q})\mathbf{D}^{-1}(\mathbf{q})(-M(\mathbf{q})[\dot{\mathbf{q}}^t\mathbf{Q}(\mathbf{q})]\mathbf{Q}(\mathbf{q})) \\ &= \mathbf{f}_Q(\mathbf{q},\dot{\mathbf{q}}) - (M(\mathbf{q})\mathbf{Q}^t(\mathbf{q})\mathbf{D}^{-1}(\mathbf{q})\mathbf{Q}(\mathbf{q}))\dot{\mathbf{q}}_Q(\mathbf{q}). \\ &= \mathbf{f}_Q(\mathbf{q},\dot{\mathbf{q}}) - \chi(\mathbf{q})\cdot\dot{\mathbf{q}}_Q(\mathbf{q}) \end{aligned} \tag{88}$$

where $\chi(\mathbf{q}) = (M(\mathbf{q})\mathbf{Q}^t(\mathbf{q})\mathbf{D}^{-1}(\mathbf{q})\mathbf{Q}(\mathbf{q}))$ is the damping coefficient of the system equation along the $\mathbf{Q}$ flow lines. Since $\mathbf{D}^{-1}(\mathbf{q})$ is positive definite, and $M(\mathbf{q})$ is a positive function, the damping coefficient is always positive (i.e., negative damping) and motion is always impeded along the flow contours of $\mathbf{Q}$ everywhere in the position space. It is easy to see that motion along a flow that is orthogonal to $\mathbf{Q}$ will not be affected by the damping control. □

VII. Results

Here, two examples are provided to demonstrate the capabilities of the proposed approach.

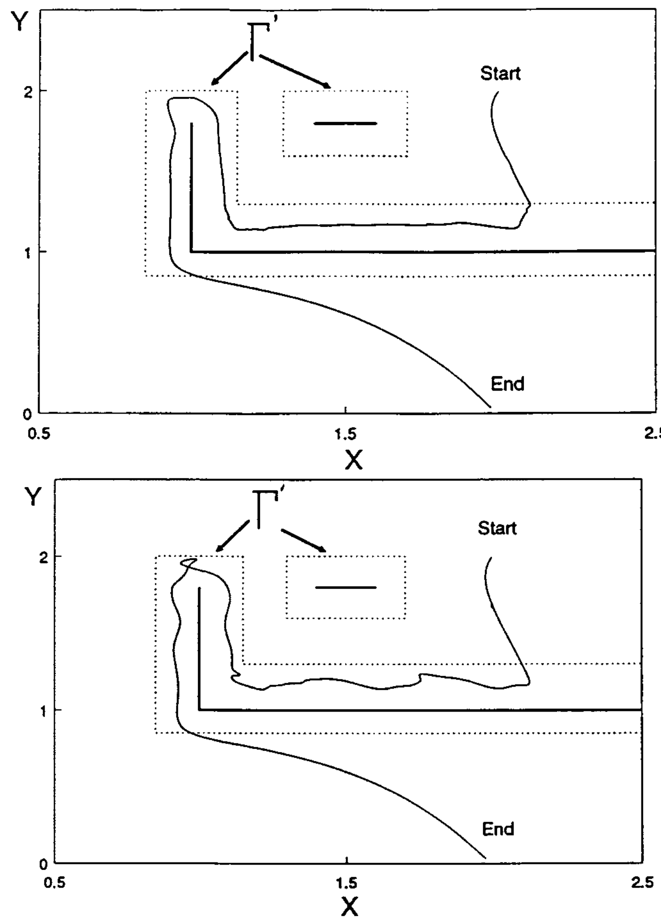


Fig. 19. (a) Robot trajectory, nonconvex obstacle, PPC, LAC, clamping, and damping controls present. (b) Removal of the damping control results in a shaky trajectory.

A. Example 1

This example demonstrates the use of the PPC to apply constraints on the position and speed of a simple second order system. The navigation control is required to drive a mass (m) along one degree of freedom the (X-axis) from an initial point $X(0) = 1$ to a final point $X(\infty) = 0$ without crossing the $X = 0$ axis. The control is also required to prevent the speed from exceeding or going below a certain specified value. It is well known that the dynamic equation for this system is a simple second order linear differential equation

$$m \cdot \ddot{X} = u \tag{89}$$

where u is the applied force and $\ddot{X}$ is the acceleration. To drive the state to its equilibrium position $(X = 0, \dot{X} = 0)$, the control law in the unconstrained state space $(u_g(X, \dot{X}))$ is taken as a simple PD controller

$$u_g(X, \dot{X}) = -[b \cdot \dot{X} + k \cdot X] \quad b > 0, \quad k > 0. \tag{90}$$

Substituting $u = u_g + u_l$, the system equation becomes

$$\ddot{X} + 2\zeta\omega_m \cdot \dot{X} + \omega_m^2 \cdot X = \frac{1}{m} u_l$$

where

$$2\zeta\omega_m = b/m \quad \text{and} \quad \omega_m^2 = k/m. \tag{91}$$

For simplicity, m and k are assumed to be equal to one. For this case ζ determines the nature of the response; if $0 < \zeta < 1$ the system is underdamped; if $\zeta = 1$ the system is critically damped; and if $\zeta > 1$ the system is overdamped.

The local component of the control (u_l) has the form

$$\begin{aligned} u_l(X, \dot{X}) &= u_{xl}(X, \dot{X}) + u_{\dot{x}l}(X, \dot{X}) \\ &= M_{xn}(X, \dot{X}) Q_n(X) + M_{\dot{x}n}(X, \dot{X}) Q_n(\dot{X}) \end{aligned} \tag{92}$$

where u_{xl} constrains the system in the position space while $u_{\dot{x}l}$ constrains the system in the velocity space. Since $Q_n(X)$ acts along one degree of freedom and is pointing in the positive direction of X,

$$Q_n(X) = 1.$$

Also

$$M_{xn}(X, \dot{X}) = \alpha_1(X, \dot{X}) V_n(X) \tag{93}$$

where

$$\nabla^2 V_n(X) = 0$$

subject to

$$V_n(0) = 1, \quad \text{and} \quad V_n(\delta_x) = 0 \quad \delta_x > 0.$$

Solving the above BVP, we have

$$V_n(X) = \left[\frac{-1}{\delta_x} \cdot X + 1\right] \quad X \in [0, \delta_x]. \tag{94}$$

Also, we have

$$\alpha_1(X, \dot{X}) = \left(\frac{k}{\delta_x}\right) |\dot{X}| \quad k = 1.0. \tag{95}$$

The resulting control has the form

$$u_{x l_n}(X, \dot{X}) = \begin{cases} \left(\frac{k}{\delta_x}\right) |\dot{X}| \cdot \left[\frac{-1}{\delta_x} \cdot X + 1\right] & X \in [0, \delta_x] \\ \text{zero} & \text{elsewhere.} \end{cases} \tag{96}$$

The PPC along $\dot{X}$ is required to prevent the speed from going below v_c $(v_c = -0.2)$; therefore, since $Q_n(\dot{X})$ is pointing in the positive direction of $\dot{X}$,

$$Q_n(\dot{X}) = 1. \tag{97}$$

Also,

$$\begin{aligned} M_{\dot{x}n}(X, \dot{X}) &= \alpha_2(X, \dot{X}) V_n(\dot{X}) \\ V_n(\dot{X}) &= \left[\frac{-1}{\delta_{\dot{x}}} \cdot (\dot{X} - v_c) + 1\right] \quad \dot{X} \in [v_c + \delta_{\dot{x}}, v_c] \\ \alpha_2(X, \dot{X}) &= (2\zeta \cdot |v_c| + |X|) \quad \delta_{\dot{x}} > 0. \end{aligned} \tag{98}$$

For the unconstrained second order system above, it can be shown that

$$|X| \leq X(0) = 1 \quad \text{for all } t. \tag{99}$$

Therefore, $\alpha_{\mathbf{2}}$ is taken as

$$\alpha_{\mathbf{2}}(X, \dot{X}) = (2\zeta \cdot |v_c| + 1). \tag{100}$$

The velocity control component has the form

$$u_{\dot{x}_{ln}}(X, \dot{X}) = \begin{cases} (2\zeta \cdot |v_c| + 1) \cdot \left[\dfrac{-1}{\delta_{\dot{x}}} \cdot (\dot{X} - v_c) + 1\right] & \dot{X} \in [v_c + \delta_{\dot{x}}, v_c] \\ \text{zero} & \text{elsewhere}. \end{cases} \tag{101}$$

In Fig. 10(a), the response in both time and phase-plane respectively, is plotted for $u_l = 0$ with $\zeta = 0.3$. In Fig. 10(b), the response is shown when the speed alone is constrained not to go below $v_c = -0.2$ at all times. Fig. 10(c) shows the response when both the position and speed are constrained not to go below $X = 0$ and $\dot{X} = -0.2$. In Fig. 11(a), only the position is constrained, the response is plotted for different δ_x, and the critically damped response of the free system $(\zeta = 1)$ is also plotted for comparison. Fig. 11(b) shows the corresponding forces. As can be seen, reducing δ_x leads to an increase in the magnitude of the decelerating force. By observing the time response, it can be noticed that the improvement in performance (in terms of the settling time) is not commensurate with the increase in the magnitude of the force.

B. Example 2

Simulation is done for a polar manipulator with only its gripper operating in the workspace. The dynamic equation for such a system is

$$\begin{bmatrix} \mathbf{Mr^2} & 0 \\ 0 & \mathbf{M} \end{bmatrix} \begin{bmatrix} \ddot{\theta} \\ \ddot{r} \end{bmatrix} + \begin{bmatrix} 2\mathbf{M}\mathbf{r}\dot{\mathbf{r}}\dot{\theta} \\ -\mathbf{M}\mathbf{r}\dot{\theta}^{\mathbf{2}} \end{bmatrix} = \begin{bmatrix} \mathbf{T} \\ \mathbf{F} \end{bmatrix} \tag{102}$$

where M is the mass ($M = 1$ kg), r is the radial distance, θ is the angle measured from the X-axis. $\mathbf{u}_g$ is a PD controller ($\mathbf{T} = kp(\theta - \theta d) + kd \cdot \dot{\theta}, F = \mathbf{k}p(r - \mathbf{r}d) + kd \cdot \dot{r}$). $kp = .5$, $kd = 3$, $\theta(0) = 45°$, $r(0) = \sqrt{8}$, $\theta d = 0$, $rd = 2$, $\dot{\theta}(0) = \dot{r}(0) = 0$, $\mathbf{u}_g = [\mathbf{T}\ \ \mathbf{F}]^t$.

Fig. 12(a) shows the path of the robot's gripper in the free space $(\mathbf{u}_l = 0)$. Fig. 12(b) and (c) show the corresponding torque and force ($\mathbf{T}$ and $\mathbf{F}$) respectively. In Fig. 13, a rectangular obstacle occupying the region $(0.6 \leq x \leq 6, 0.8 \leq y \leq 1.2)$ is placed in the path of the arm. To prevent collision, a PPC is placed around the obstacle in a surrounding rectangular region of minimum width $\delta = 0.1$. The strength of the PPC is set to zero at the outer boundary $\Gamma\delta$ and is set to the maximum value at the obstacle boundary (Γ). The PPC was constructed without the $O\delta_d$ region. This is made possible by making sure that the average strength of the PPC in $O\delta$ satisfies (31).

In Fig. 13, the radial force field successfully prevented the gripper from colliding with the obstacle. However, the motion bounced back and forth on the obstacle's surface until it finally settled short of reaching its target. Fig. 14 shows the response when an additional control is used to confine the gripper to an *a priori* specified region around the obstacle that has a boundary Γ' (clamping control). The minimum distance between $\Gamma\delta$ and Γ' is set to $\delta_d = 0.1$. Such a control reduced the magnitude of the oscillations and confined the motion to an *a priori* known region.

To further reduce the oscillations, a control field is placed between Γ and Γ' to damp the motion along the normal flow lines to the obstacle's surface (Fig. 15). This control component allows for a steady path around the obstacle while enabling the motion to slide unimpeded along the obstacle's surface. Such an approach does not slow down the system unlike the case in which path smoothness is achieved by increasing the damping term of the PD controller. In addition to improving the quality of the path, the damping control results in a well-behaved torque and force waveforms that have lower magnitudes and less energy than those in which damping is not present.

In Fig. 16, an LAC is added between Γ and Γ' with a strength that is set to zero at Γ'. The clamping control and the damping control are present. The LAC yanked the arm from the local equilibrium zone and drove it around the obstacle so that $\mathbf{u}_g$ is able to sweep it to the target. In Fig. 17, anisotropic damping is removed resulting in a shaky path. Also, the quality of the control signal has deteriorated, with an increase in the peak magnitude of the control signal as well as the appearance of oscillations. This increases the strain on the robot's actuators. It also increases energy consumption.

In Fig. 18, the clamping control is also removed. As a result, the field from $\mathbf{u}_g$ pushed the arm outside the region of efficacy of the LAC, thereby, trapping the robot in a local minimum. Fig. 19(a) demonstrates the decoupled nature of the suggested control and its ability to handle nonconvex regions. The presence of the small rectangular obstacle did not at all interfere with the operation of the steering control of the nonconvex obstacle. This enables the designer to remove it or change its location without having to worry about the effect that this might have on the other steering controls in the workspace. In Fig. 19(b), the anisotropic damping component of the control is removed, yielding a shaky path.

VIII. CONCLUSION

In this paper a method is suggested for applying constraints on the state of a robot manipulator using the artificial vector potential approach. The path planning problem considered in this work enables the robot to be driven along a well-behaved and safe path to a desired destination. Such a task is performed through a special kind of control called the navigation control (NC). In effect, this control functions to provide the robot with a goal-oriented awareness of its environment. The NC is designed so that the effort needed to adjust the control following a change in the geometry of the environment be proportional to that change. This design enables the construction of a set of behavioral primitives that consists of ready-to-use global fields, each designed to perform an *a priori* specified task where change in the environment can be, with reasonable effort, accounted for.

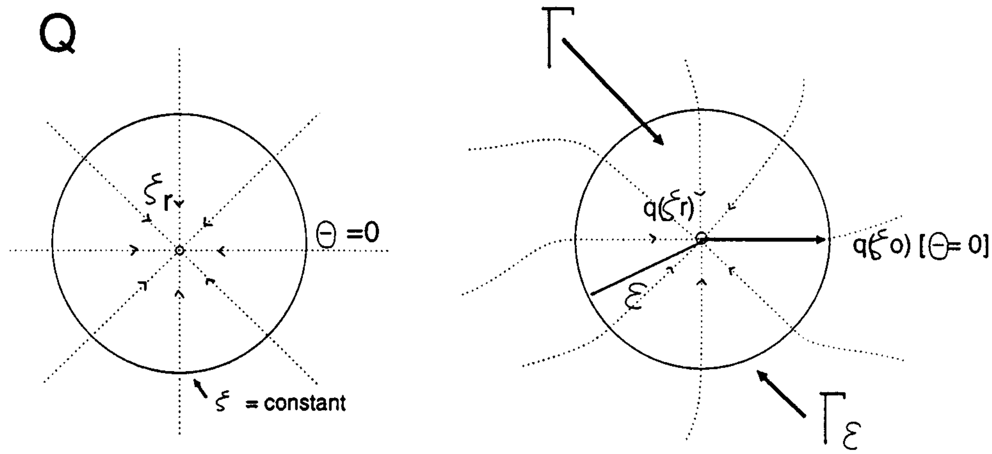


Fig. 20. Field lines near the exit point on the surface of the obstacle (Γ) and its image Q.

The suggested new approach for NC synthesis is necessary to avoid the difficulties encountered by the past approaches. In particular, the local decentralized strategy to navigation, the superior steering capabilities of a VPF, the flexibility of a BVP formulation, and the response conditioning of the anisotropic damping control are keys to the success of the proposed NC approach.

APPENDIX I

Here, a procedure that uses $V_{\mathbf{1}t}(\mathbf{q})$ is suggested for mapping a given ξ to the corresponding $\mathbf{q}(\xi)$, and vice versa. Since $V_{\mathbf{1}t}$ is a Harmonic (and in turn analytic) function, the mapping which is defined by it is conformal (i.e., angle-preserving) except when its derivative is zero (e.g., at $\mathbf{q}(\xi_r)$ [72, p. 565])). This property is used for specifying an angle (θ) for the vector ξ. First, let Γ_ε be a tiny sphere in Γ with $\mathbf{q}(\xi_r)$ as its center (Fig. 20)

$$\Gamma_\varepsilon = \{\mathbf{q}\colon |\mathbf{q} - \mathbf{q}(\xi\varepsilon)|\} \quad \varepsilon \to 0, \quad q \in \Gamma. \tag{103}$$

For a very small ε, the gradient flow lines of $V_{\mathbf{1}t}$ inside $\Gamma\varepsilon$ have the same configuration as those of ξ in Q (assuming a differentiable Γ). This makes it possible to assign to ξ the angle at $\mathbf{q} \in \Gamma\varepsilon$ which is measured from an arbitrarily chosen reference position $\mathbf{q}(\xi_o) \in \Gamma\varepsilon$. Given a ξ (both magnitude $(|\xi|)$ and angle (θ)), the corresponding $\mathbf{q}(\xi)$ can be computed by first choosing a $\mathbf{q}(\xi_s) \in \Gamma\varepsilon$ such that

$$\mathrm{Arg}(\nabla V_{\mathbf{1}t}(\mathbf{q}(\xi_s))) = \theta(\xi) \tag{104}$$

then using the differential equation

$$\dot{\mathbf{q}} = \nabla V_{\mathbf{1}t}(\mathbf{q}) \quad \mathbf{q}(0) = \mathbf{q}(\xi_s) \tag{105}$$

to traverse a path (ρ) on Γ that has a length equal to $|\xi|$. The end point of $\rho(\mathbf{q}(\xi_f))$ is the point of interest that corresponds to the given ξ. Vice versa, given a $\mathbf{q}(\xi_f)$, the corresponding $\mathbf{x}$ can be computed by traversing a path ρ using the differential equation

$$\dot{\mathbf{q}} = -\nabla V_{\mathbf{1}t}(\mathbf{q}) \quad \mathbf{q}(0) = \mathbf{q}(\xi_f). \tag{106}$$

This path is made to terminate at $\mathbf{q}(\xi_s) \in \Gamma\varepsilon$. The length of ρ is taken as $|\xi|$, and the $\mathrm{Arg}(\nabla V_{\mathbf{1}t}(\mathbf{q}(\xi_s)))$ is equal to $\theta(\xi)$.

REFERENCES

[1] A. Masoud and S. Masoud, "Evolutionary action maps for navigating a robot in an unknown, multidimensional, stationary, environment, part II: Implementation & results," in *IEEE Int. Conf. Robotics and Automation*, NM, Apr. 21–27, 1997, pp. 2090–2096.

[2] ——, "A self-organizing, hybrid, PDE-ODE structure for motion control in informationally-deprived situations," in *37th IEEE Conf. Decision and Control*, Tampa, FL, Dec. 16–18, 1998, pp. 2535–2540.

[3] A. Masoud, "Using hybrid, vector-harmonic potential fields for multi-robot, multi-target navigation in a stationary environment," in *IEEE Int. Conf.Robotics and Automation*, Minneapolis, MN, Apr. 22–28, 1996, pp. 3564–3571.

[4] A. Masoud, "Evolutionary Action Maps for Navigating a Robot in an Unknown, Multi-Dimensional, Stationary Environment, Part I: The Navigator's Structure,", Internal paper.

[5] A. Masoud and S. Masoud, "A modified, hybrid, PDE-ODE controller with integrated directional and region avoidance constraints," in *37th IEEE Conf. Decision and Control*, Tampa, FL, Dec. 16–18, 1998, pp. 2021–2022. An extended version of the paper appeared in the IEEE International Conference on Robotics and Automation, San Franscisco, CA, April 22–28, 2000.

[6] A. Masoud, "Techniques in Potential-Based Path Planning," Ph.D. Thesis, Elect. Comput. Eng. Dept., Queen's Univ., Kingston, Ont., Canada, Mar. 1995.

[7] C. Hull, "The goal gradient hypothesis and maze learning," *Psychol. Rev.*, vol. 39, no. 1, pp. 25–43, Jan. 1932.

[8] ——, "The goal-gradient hypothesis applied to some 'Field-Force' problems in the behavior of young children," *Psychol. Rev.*, vol. 45, no. 4, pp. 271–299, July 1938.

[9] L. Loef, "An Algorithm for Computer Guidance of a Manipulator in Between Obstacles," M.S. thesis, Oklahoma State Univ., Stillwater, 1973.

[10] L. Loef and A. Soni, "An algorithm for computer guidance of a manipulator in between obstacles," *Trans. ASME, J. Eng. Ind.*, pp. 836–842, Aug. 1975.

[11] O. Khatib, "Real-time obstacle avoidance for manipulators and mobile robots," in *IEEE Int. Conf. Robotics and Automation*, St. Louis, MO, Mar. 25–28, 1985, pp. 500–505.

[12] B. Krogh, "A generalized potential field approach to obstacle avoidance control," in *Robotics Research: The Next Five Years and Beyond*, Bethlehem, PA, USA, Aug. 14–16, 1984, pp. 1–15.

[13] ——, "Guaranteed steering control," in *The American Control Conf.*, Boston, MA, June 19–21, 1985, pp. 950–955.

[14] M. Takegaki and S. Arimoto, "A new feedback method for dynamic control of manipulators," *Trans. ASME J. Dyn. Syst. Meas. Control*, vol. 102, pp. 119–125, June 1981.

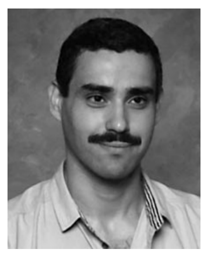
[15] V. Pavlov and A. N. Voronin, "The method of potential functions for coding constraints of the external space in an intelligent mobile robot," *Sov. Automat. Contr.*, vol. 17, no. 6, pp. 45–51, 1984.

[16] J. Andrews and N. Hogan, "Impedance control as a framework for implementing obstacle avoidance in manipulator," in *Control of Manufacturing Processes and Robotics Systems*, D. Hardt and W. Book, Eds, pp. 243–251. Also presented at the Winter Annual Meeting of the American Society of Mechanical Engineers, Boston, Massachusetts, Nov. 13–18, 1983.
[17] O. Khatib, "The operational space formulation in the analysis, design, and control of robot manipulators," in *Robotics Research, 3rd Int. Symp.*, O. Faugeras and G. Giralt, Eds. Cambridge, MA: MIT Press, 1986, pp. 263–270.
[18] W. Newman and N. Hogan, "High speed robot control and obstacle avoidance using dynamic potential functions," in *Proc. 1987 IEEE Int. Conf. Robotics and Automation*, Raleigh, NC, Mar. 31–Apr. 3 1987, pp. 801–805.
[19] R. Tilove, "Local obstacle avoidance for mobile robots based on the method of artificial potentials," in *IEEE Int. Conf. Robotics and Automation*, Cincinnati, OH, May 13–18, 1990, pp. 566–571.
[20] R. Aggarwal and G. Leitmann, "Avoidance control," in *Trans. ASME, J. Dyn. Syst. Meas. Contr.*, June 1972, pp. 152–154.
[21] ——, "A maxmin distance problem," *Trans. ASME, J. Dyn. Syst., Meas. Contr.*, pp. 155–158, June 1972.
[22] B. Barmish, W. Schmitendorf, and G. Leitmann, "A note on avoidance control," *Trans. ASME, J. Dyn. Syst., Meas., Contr.*, vol. 103, pp. 69–70, Mar. 1981.
[23] G. Leitmann and J. Skowronski, "Avoidance control," *J. Optim. Theory Appl.*, vol. 23, no. 4, pp. 581–591, Dec. 1977.
[24] G. Leitmann, "Guaranteed avoidance feedback control," *IEEE Trans. Automat. Contr.*, vol. AC-25, pp. 850–851, Aug. 1980.
[25] W. Schmitendorf, B. Barmish, and B. Elenbogen, "Guaranteed avoidance control and holding control," *Trans. ASME, J. Dyn. Syst., Meas., Contr.*, vol. 104, pp. 166–172, June 1982.
[26] D. Koditschek, "Exact robot navigation by means of potential functions: Some topological considerations," in *IEEE Int. Conf. Robotics and Automation*, Raleigh, NC, Mar. 1987, pp. 1–6.
[27] E. Rimon and D. Koditschek, "Exact robot navigation using cost functions: The case of distinct spherical boundaries in R^n," in *Proc. 1988 IEEE Int. Conf. Robotics and Automation*, Philadelphia, PA, Apr. 24–29, pp. 1791–1796.
[28] ——, "The construction of analytic diffeomorphisims for exact robot navigation on star worlds," in *Proc. 1989 IEEE Int. Conf. Robotics and Automation*, Scottsdale, AZ, May 14–19, pp. 21–26.
[29] ——, "Robot navigation functions on manifolds with boundary," *Adv. Appl. Math.*, vol. 11, pp. 412–442, 1990.
[30] ——, "Exact robot navigation in geometrically complicated but topologically simple spaces," in *Proc. 1990 IEEE Int. Conf. Robotics and Automation*, Cincinnati, OH, May 13–18, pp. 1937–1942.
[31] ——, "Exact robot navigation using artificial potential functions," *IEEE Trans. Robot. Automat.*, vol. 8, pp. 501–518, Oct. 1992.
[32] D. Koditschek, "The application of total energy as a liapunov function for mechanical control systems," in *Dynamics and Control of Multibody Systems, Contemporary Mathematics, Proc. Summer Research Conf.*, vol. 97, July 30–Aug. 5 1988, 1989, pp. 131–157.
[33] ——, "The control of natural motion in mechanical systems," *Trans. ASME, J. Dyn. Syst., Meas., Contr.*, vol. 113, pp. 547–551, Dec. 1991.
[34] S. Sundar and Z. Shiller, "Near-time optimal path planning using potential functions," in *IEEE/RSJ Int. Workshop on Intelligent Robots and Systems, IROS'91*, Osaka, Japan, Nov. 3–5, 1991, pp. 1269–1274.
[35] K. Satoh, "Collision avoidance in multi-dimensional space using laplace potential," in *Preprint of 5th Annual Conf. Robotics Society of Japan*, 1987, pp. 155–156. in Japanese.
[36] ——, "Deadlock-free motion planning using the laplace potential field," *Adv. Robotics*, vol. 7, no. 5, pp. 449–461, 1993.
[37] C. Connolly, R. Weiss, and J. Burns, "Path planning using laplace's equation," in *IEEE Int. Conf. Robotics and Automation*, Cincinnati, OH, May 13–18, 1990, pp. 2102–2106.
[38] E. Prassler, "Electrical networks and a connectionist approach to path finding," in *Connectionism in Perspective*, R. Pfeifer, Z. Schreter, F. Fogelman, and L. Steel, Eds. Amsterdam, The Netherlands: Elsevier, 1989, pp. 421–428.
[39] G. Lei, "A neuron model with fluid properties for solving labyrinthine puzzle," *Biol. Cybern.*, vol. 64, pp. 61–67, 1990.
[40] E. Plumer, "Cascading a systolic and a feedforward neural network for navigation and obstacle avoidance using potential fields," prepared for Ames Research Center, Contract NGT-50 642, NASA Contractor Rep. 177 575, Feb. 1991.
[41] J. Decuyper and D. Keymeulen, "A reactive robot navigation systems based on a fluid dynamics metaphor," in *Lecture Notes in Computer Science (496)*, G. Goos and J. Hartmanis, Eds: Proceedings, Springer-Verlag, Oct. 1–3, 1990, pp. 356–362. Parallel Problem Solving from Nature, 1st workshop, PPSN1, Dortmund, FRG.
[42] D. Keymeulen and J. Decuyper, "The fluid dynamics applied to mobile robot motion: The stream field method," in *IEEE Int. Conf. Robotics and Automation*, San Diego, CA, May 8–13, 1994, pp. 378–385.
[43] S. Akishita, S. Kawamura, and K. Yayashi, "New navigation function utilizing hydrodynamic potential for mobile robot," in *IEEE Int. Workshop on Intelligent Motion Control*, Istanbul, Turkey, Aug. 20–22, 1990, pp. 413–417.
[44] S. Akishita, S. Kawamura, and T. Hisanobu, "Velocity potential approach to path planning for avoiding moving obstacles," *Adv. Robot.*, vol. 8, no. 3, pp. 338–349, June 1992.
[45] J. Kim and K. Khosla, "Real-time obstacle avoidance using harmonic potential functions," *IEEE Trans. Robot. Automat.*, vol. 8, pp. 338–349, June 1992.
[46] L. Tarassenko and A. Blake, "Analogue computation of collision-free paths," in *IEEE Int. Conf. Robotics and Automation*, Sacramento, CA, Apr. 1991, pp. 540–545.
[47] Z. Li and T. Bui, "Robot path planning and navigation using fluid model," in *2nd Int. Conf. Automation, Robotics, and Computer Vision*, Sept. 16–18, 1992, pp. RO-10.3.1–RO-10.3.5.
[48] D. Jones and J. Gonzalez, "A flow field approach to planning robot paths in parallel," in *Int. Conf. Control'94*, Convorny, UK, Mar. 21–24, pp. 1111–1116.
[49] K. Althofer, D. Fraser, and G. Bugmann, "Rapid path planning for robotic manipulators using an emulated resistive grid," *Electron. Lett.*, vol. 31, no. 22, pp. 1960–1961, Oct. 26, 1995.
[50] D. Megherbi and W. Wolovich, "Modeling and automatic real-time motion control of wheeled mobile robots among moving obstacles: Theory and applications," in *32nd Conf. Decision and Control*, San Antonio, TX, Dec. 15–17, 1993, pp. 2676–2681.
[51] ——, "Real-time velocity feedback obstacle avoidance via complex variables and conformal mapping," in *IEEE Int. Conf. Robotics and Automation*, Nice, France, May 1992, pp. 206–213.
[52] A. Masoud and S. Masoud, "Robot navigation using a pressure generated mechanical stress field, the biharmonic potential approach," in *IEEE Int. Conf. Robotics and Automation*, San Diego, CA, May 8–13, 1994, pp. 124–129.
[53] G. Schmidt and K. Azarm, "Mobile robot navigation in a dynamic world using an unsteady diffusion equation strategy," in *Proc. IEEE/RSJ Int. Conf. Intelligent Robots and Systems*, Raleigh, NC, July 7–10, 1992, pp. 642–647.
[54] A. Masoud, "A boundary value problem formulation of pursuit-evasion in a stationary environment: A potential field approach," in *IEEE Int. Conf. Robotics and Automation*, Nagoya, Japan, May 21–27, 1995, pp. 2734–2739.
[55] J. Guldner and V. Utkin, "Tracking the gradient of artificial potential fields: Sliding mode control for mobile robots," *Int. J. Contr.*, vol. 63, no. 3, pp. 417–432, 1996.
[56] V. Utkin, S. Drakunov, H. Hashimoto, and F. Harashima, "Robot path obstacle avoidance control via sliding mode approach," in *IEEE/RSJ Int. Workshop on Intelligent Robots and Systems IROS91*, Osaka, Japan, Nov. 3–5, 1991, pp. 1287–1290.
[57] J. Guldner and V. Utkin, "Sliding mode control for an obstacle avoidance strategy based on an harmonic potential field," in *Proc. 23rd Conf. Decision and Control*, San Antonio, TX, Dec. 15–17, 1993, pp. 424–429.
[58] J. Guldner, V. Utkin, H. Hashimoto, and F. Harashima, "Obstacle avoidance in R^n based on artificial harmonic potential fields," in *IEEE Int. Conf. Robotics and Automation*, Nagoya, Japan, May 21–27, 1995, pp. 3051–3056.
[59] J. Guldner and V. Utkin, "Sliding mode control for gradient tracking and robot navigation using artificial potential fields," *IEEE Trans. Robot. Automat.*, vol. 11, pp. 247–254, Apr. 1995.
[60] C. Connolly and R. Grupen, "The application of harmonic functions to robotics," *J. Robot. Syst.*, vol. 10, no. 7, pp. 931–946, 1993.
[61] C. Connolly, K. Souccar, and R. Grupen, "A hamiltonian framework for kinodynamic planning and control," in *1995 IEEE Int. Conf. Robotics and Automation*, Nagoya, Japan, May 21–27, 1995, pp. 2746–2751.
[62] P. Morse and H. Feshbach, *Methods of Theoretical Physics*. New York: McGraw-Hill, 1953, vol. 1–2.
[63] S. Arimoto and F. Miyazaki, "Stability and robustness of PID feedback control for robotics manipulators of sensory capabilities," in *Proc. 1st Int. Symp. Robotics Research*, 1984, pp. 385–407.

[64] D. Koditschek, "Natural motion for robot arms," in *23rd Conf. Decision and Control*, Las Vegas, NV, Dec. 1984, pp. 733–735.
[65] J. R. Hanna, *Fourier Series and Integrals of Boundary Value Problems*. New York: Wiley, 1982.
[66] O. Zienkiewicz and K. Morgan, *Finite Element and Approximation*. New York: Wiley, 1983.
[67] S. Hoole, *Computer-Aided Analysis & Design of Electromagnetic Devices*. Amsterdam, The Netherlands: Elsevier, 1989.
[68] V. P. Howen, "Finite difference method for solving partial differential equations," in *Amsterdam Math. Centr.*, 1968.
[69] I. Stakgold, *Green's Functions and Boundary Value Problems*. New York: Wiley, 1979.
[70] C. Brebbia, *The Boundary Element Method for Engineers*, 2nd ed. New York: Pentech Press, 1984.
[71] C. Brebbia, J. Telles, and L. Worbel, *Boundary Element Techniques, Theory and Applications in Engineering*, Berlin: Springer-Verlag, 1984.
[72] E. Kreyszig, *Advanced Engineering Mathematics*. New York: Wiley, 1967.
[73] A. Manz, R. Liscano, and D. Green, "A comparison of real-time obstacle avoidance methods for mobile robots," in *Experimental Robotics II, the 2nd International Symposium*, R. Chatila and G. Hirzinger, Eds, Toulouse, France: Springer-Verlag, June 25–27, 1991, pp. 301–316.
[74] Y. Koren and J. Borenstein, "Potential field methods and their inherent limitations for mobile robot navigation," in *Proc. 1991 IEEE Int. Conf. Robotics and Automation*, Sacremento, CA, Apr. 1991, pp. 1398–1404.
[75] C. McInnes, "Autonomous proximity maneuvering using artificial potential functions," *Eur. Space Agency J.*, vol. 17, pp. 159–169, 1993.
[76] R. Volpe, "Techniques for collision prevention, impact stability, and force control by space manipulators," in *Teleoperation and Robotics in Space*, S. Skaar and C. Ruoff, Eds. Washington, DC: AAAI, 1994, pp. 175–208.
[77] P. Eiseman, "A multi-surface method of coordinate generation," *J. Comput. Phys.*, pp. 118–150, 1979.
[78] C. Mobley and R. Stewart, "On the numerical generation of boundary-fitted orthogonal curvilinear coordinate systems," *J. Comput. Phys.*, pp. 124–135, 1980.
[79] W. Newman, "Robust near time-optimal control," *IEEE Trans. Automat. Contr.*, vol. 35, pp. 841–844, July 1990.
[80] W. Newman and N. Hogan, "Time optimal control of balanced manipulators," *Trans. ASME: J. Dyn. Syst., Meas., Contr.*, vol. 111, pp. 187–193, June 1989.
[81] W. Newman and K. Souccar, "Robust, near time-optimal control of nonlinear second-order systems: Theory and experiments," *Trans. ASME J. Dyn., Syst., Meas. Contr.*, vol. 113, pp. 363–370, Sept. 1991.
[82] G. Strang, *Linear Algebra and its Applications*, 2nd ed. New York: Academic, 1980.
[83] J. Synge and A. Schild, *Tensor Calculus*. Toronto, Ont., Canada: Univ. Toronto Press, 1949.
[84] F. Bullo, "Stabilization of relative equilibria for systems on riemannian manifolds," in *Proc. American Control Conf.*, San Diego, CA, June 1999, pp. 1618–1622.

Samer A. Masoud received the B.Sc. degree in mechanical engineering and the M.Sc. degree in mechanical engineering (applied mechanics) from Jordan University of Science and Technology, Irbrd, Jordan, in 1987 and 1990, respectively, and the Ph.D. degree in applied mechanics from the Mechanical Engineering Department, University of Cincinnati, OH, in 1994.

He is currently an Assistant Professor at the Mechanical Engineering Department, Jordan University of Science and Technology. His current interests include design, mechanisms, vibration, and robotics.



Ahmad A. Masoud received the B.Sc. degree in power systems from the Electrical Engineering Department, Yarmouk University, Jordan, in 1985, and the M.Sc. (digital signal processing) and Ph.D. degrees (robotics) both in electrical engineering, in 1989 and 1995, respectively, from Yarmouk University.

He was a Researcher at the Electrical Engineering Department, Jordan University of Science and Technology from 1985 to 1987. He also worked as a part-time Assistant Professor in the Electrical Engineering Department, Royal Military College of Canada, from 1996 to 1998. His current interests include navigation, planning, constrained intelligent motion control, and wireless communication.