December 2004

Feedback-Based Event-Driven Parts Moving

Cem Serkan Karagöz
Ford Motor Company

H. Isil Bozma
Boğaziçi University

Daniel E. Koditschek
University of Pennsylvania, kod@seas.upenn.edu

Feedback-Based Event-Driven Parts Moving

Abstract

A collection of unactuated disk-shaped "parts" must be brought by an actuated manipulator robot into a specified configuration from arbitrary initial conditions. The task is cast as a noncooperative game played among the parts—which in turn yields a feedback-based event-driven approach to plan generation and execution. The correctness of this approach, an open question, has been demonstrated in simpler settings and is further suggested by the extensive experiments reported here using an actual working implementation with EDAR—a mobile robot operating in a purely feedback-based event-driven manner. These results verify the reliability of this approach against uncertainties in sensory information and unanticipated changes in workspace configuration.

Keywords

artificial potential functions, autonomous robots, game theory, parts rearrangement

Comments

NOTE: At the time of publication, author Daniel Koditschek was affiliated with the University of Michigan. Currently (August 2005), he is a faculty member in the Department of Electrical and Systems Engineering at the University of Pennsylvania.

Feedback-Based Event-Driven Parts Moving

Cem Serkan Karagöz, H. Işıl Bozma, and Daniel E. Koditschek

Abstract—**A collection of unactuated disk-shaped "parts" must be brought by an actuated manipulator robot into a specified configuration from arbitrary initial conditions. The task is cast as a noncooperative game played among the parts—which in turn yields a feedback-based event-driven approach to plan generation and execution. The correctness of this approach, an open question, has been demonstrated in simpler settings and is further suggested by the extensive experiments reported here using an actual working implementation with EDAR—a mobile robot operating in a purely feedback-based event-driven manner. These results verify the reliability of this approach against uncertainties in sensory information and unanticipated changes in workspace configuration.**

Index Terms—**Artificial potential functions, autonomous robots, game theory, parts rearrangement.**

I. Introduction

This paper addresses a geometrically simplified version of the warehouseman's problem [15], as depicted in Fig. 1. A two-dimensional (2-D) workspace contains an actuated robot and a set of unactuated rigid body parts. Each part, although unactuated, is free to move on the plane if it is towed by the robot. Furthermore, each part has a stationary goal location specified by the user. The ensemble of these locations specifies the robot's task: since the bodies have no actuators and cannot reposition themselves, the robot must move each one to its respective goal. We now depart from the classical warehouseman's paradigm to introduce the new requirement that the robot's control strategy be feedback-based rather than a tracking strategy in support of an *a priori* (open loop) plan of motion. The robot's plan must be feedback-based because there is no guarantee that the bodies' relative placements will be left undisturbed. The plan must be correct in that it should be guaranteed eventually to shepherd all of the parts to their specified destinations. Or, if the goal configuration is infeasible, that is, if the initial placement is not path connected to the goal in the problem configuration space, then the plan must guarantee that the robot terminates with a return to a specified "nest" (rather than allowing useless cycling from piece to piece).

A. Contributions of the Paper

The contribution of the present paper is twofold. First, we generalize the feedback-based event-driven formulation of the parts moving problem to the case of 2 degrees of freedom (DOF).[1] Second, we report on an extensive and systematic experimental assessment using EDAR—the first physical implementation of a purely feedback-based event-driven parts moving scheme on a mobile robot. By "feedback-based event-driven," we mean a hybrid dynamical control system that is controlled purely by state feedback. In this approach, planning and action phases are unified: at each state encountered along the way, the robot's motion plans and control commands are generated simultaneously by a family of closed-loop vector fields. The formal algorithm is predicated on the assumptions that the robot: 1) has perfect (online) knowledge of the size and locations of the parts; 2) knows the goal positions of the parts; 3) has ideal bounded torque actuators; and 4) has perfect (online) knowledge of its joint positions. As usual, in actual implementation, these assumptions do not exactly apply since the relevant information is fed back from the optical encoders and camera-based vision system. Despite the considerable sensor inaccuracy, EDAR performs nearly in real-time, reliably, and with accuracy sufficient to the global "warehouseman's" task. We do not have at present the desired formal proof of correctness for this hybrid system, however, previous constructions of a similar (albeit much simpler) nature [5], [19], [27] have been successfully proven correct. In our view, prior examples of correct algorithms along with the success of this experimental study provide a very strong motivation for extending the theoretical results to the present setting—which despite the simplicity of the problem statement—has proven to be complicated [17].

B. Feedforward and Feedback Plans

Traditionally, the warehouseman's problem is solved by open loop plans via a sequence of task planning, trajectory generation, and execution stages [13], [21], [23], [29]. If the environment is time varying—even if the change is slow and sometimes intermittent as is the case in many realistic settings—there is a risk of these plans becoming obsolete very quickly at the cost of growing inefficiency and, ultimately, as is often the case, failure [6]. In such situations, solutions to any motion-planning problem will need to be reactive, and in practical implementations hybrid strategies are typically introduced [9], [11], [28].

Fig. 1. EDAR (the robot) and the parts (vertical white cylinders).

Manuscript received March 13, 2003. This paper was recommended for publication by Associate Editor Amato and Editor S. Hutchinson upon evaluation of the reviewers' comments. This work was supported by NSF-TÜBİTAK INT-9819890 and TÜBİTAK MİSAG65. This paper was presented in part at the IEEE International Conference on Robotics and Automation, San Francisco, CA, April 2000.

C. S. Karagöz was with the Intelligent Systems Laboratory, Bogazici University, Bebek 34342 Istanbul, Turkey. He is now with the Manufacturing Engineering, European Engines, Ford Motor Company, Ford Werke AG (NW/E-041), D-50725 Koeln, Germany (e-mail: ckaragoz@ford.com).

H. I. Bozma is with the Intelligent Systems Laboratory, Electric Electronic Engineering Department, Bogaziçi University, Bebek 34342 Istanbul, Turkey (e-mail: bozma@boun.edu.tr).

D. E. Koditschek is with the Artificial Intelligence Laboratory, Electrical Engineering and Computer Science Department, College of Engineering, University of Michigan, Ann Arbor, MI 48109-2110 USA (e-mail: kod@umich.edu).

Digital Object Identifier 10.1109/TRO.2004.829503

[1]This formulation, although an extension of the one-dimensional (1-D) case [5], introduces quite a number of additional details since the number of parts is no longer restricted to two and the robot's switching mechanism must be more sophisticated than a mere alternation between the two parts.

One view of how much harder than obstacle avoidance is the warehouseman's problem of present interest obtained from relatively recent asymptotic complexity analysis of general manipulation planning and rearrangement problems [2], [8], [22], [26]. It has been understood for nearly two decades that such problems are PSPACE-hard in settings of the kind considered here—2-D environments where only translations are allowed and where the task specification explicitly includes the final positions of all movable objects [15], [32]. Perhaps because of the known complexity, even (open-loop) planning results have been few, such as for the case of a polygonal robot and one polygonal movable obstacle in a polygonal environment of complexity $O(n)$ with $O(n^3 \log^2 n)$ preprocessing and $O(n^2)$ time complexity [32]. To the best of the authors' knowledge, no complete open-loop algorithms for the case of multiple movable obstacles have yet been formulated. The most constructive approaches to this task domain have been simulation-based—a notable exception provided by [2] and [3] who have reported implementing their algorithm on a mobile robot equipped with perfect world information and no sensing. The importance of incorporating reactivity in rearrangement planning has been acknowledged as well [25].

Research focused on the extreme opposite paradigm that we embrace here—purely feedback-based event-driven algorithms—has been much less popular. No doubt the relative rarity of work in this area is due in part to the perception that it is intractable. However, for the simpler stationary obstacle avoidance problem where a single robot is the only movable component of the workspace and a set of fixed obstructions must be circumnavigated to some selected destination, a variety of potential field heuristics have been presented in the literature [7], [10], [14], [18], [24], [30]. Furthermore, for suitably simplified versions of that problem, it has been possible to construct functions which guarantee that the moving robot will avoid obstacles and arrive at the desired destination [19], [27]. These developments motivate our feeling that implementation and analysis of the reactive version of the warehouseman's problem may also be effectively pursued. A recent series of papers documents a slowly growing set of analytical results for simplified versions. Here, geometric simplification arising from the restriction to parts and robots with a perfectly circular footprint enforces the focus on the key issue that distinguishes the approach: state and sensory event-driven reactive planning. The last author has studied the simple problem of a point robot on a wire required to move a collection of unactuated beads on a parallel wire, introducing a navigation function [27] (a refined notion of artificial potential functions) to encode the participation of each part in the completed task with formal guarantees of success [19]. In a subsequent paper [5], a slightly generalized version of the problem was considered wherein the point robot inhabits the same copy of the part workspace $\mathbf{R}$ instead of a parallel isolated copy. The formulation of the problem as a noncooperative game played among the parts with the robot as a "referee" results in a plan generated completely in a reactive manner and is ensured of a schedule of matings resulting either in task completion or termination if the task is not feasible. In the present work, we generalize the scheme in [5] to the planar setting, maintaining the guaranteed obstacle avoidance, but losing the proof of global convergence. Notwithstanding the absence of such a proof, past and present experience strongly suggests that the basin of attraction of the goal configuration induced by the closed-loop system can be made an arbitrarily large fraction of the free configuration space volume.

II. A Noncooperative Game

The essential nonholonomic nature of the parts moving problem precludes the possibility of smooth feedback stabilization, leading to the hybrid scheme of switched smooth flows introduced in [19] and generalized here. When the robot inhabits the same workspace as the parts, it is most natural to interpret these switched flows as "moves" in a noncooperative dynamical game [1], reflecting the conflict between individual convergence and collective pattern formation as articulated in [5]. Parts become players at the lower level of "moves" in the workspace. At the higher level, the robot acts as a referee of the game played among the parts—it decides which player gets to move next and then moves it accordingly. The term game is used to describe the discrete dynamical system on the state space of parts—as will be formulated in the sequel.

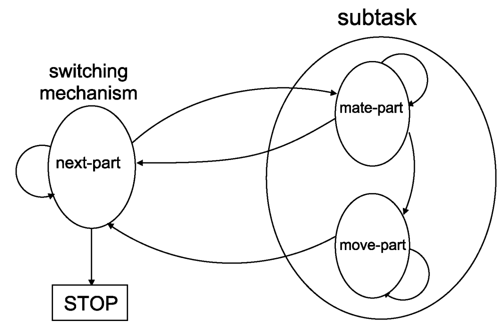

Fig. 2. Game automaton—the transitions are labeled according to (4).

Each part $i \in P = \{1, \ldots, p\}$, $p \in \mathbf{Z}^+$, is defined by its center $b_i \in \mathbf{R}^2$ and radius $\rho_i \in \mathbf{R}$. The state vector of all of the parts $b \in R^{2p}$ is defined as $b \triangleq \sum_{i \in P} b_i \otimes e_i$, where $e_1, e_2, \ldots, e_n \in \mathbf{R}^p$ are the unit vectors in $\mathbf{R}_p$. The goal of each player i is to be moved to its goal position $g_i \in \mathbf{R}^2$ without colliding with other players. The vector of all the goal positions $g \in \mathbf{R}^{2p}$ is defined as $g \triangleq \sum_{i \in P} g_i \otimes e_i$. The robot is defined by its center $r \in \mathbf{R}^2$, angle of its gripper with respect to the x axis θ and radius $\rho_r \in \mathbf{Z}^+$. The augmented robot state vector is defined as $r_a \in \mathrm{SE}(2)$ as $r_a \triangleq [r \;\; \theta]^T$.

The rules of the game are as follows.

- The robot should mate with a part before moving it and both motions must be free of collisions.
- The robot can move one part at a time.
- An urgency measure is used by the robot in order to decide which part to move next.

From the robot's perspective, the first rule presupposes the availability of a collection of p subgoal strategies (that ensure mating and moving a particular part without collision). According to the second rule, one subtask gets to be executed at a time. Finally, the third rule implies that the subtasks are competing and the robot selects the subtask based on an urgency measure. The resulting game can be described by an automaton, as shown in Fig. 2.

As the robot is included in the workspace as a body with physical extent, the workspace has a different geometry depending on whether the robot is moving to grasp a part (mating) or actually carrying a part (moving). Consequently, two control laws are required for each subtask. The first control law $\mathrm{mate\_part}$ enables robot-part mating. The second control law enables the robot to carry this part to its destination or an intermediate position. Successful completion of the task depends on the intrinsic properties of the game played among the players—namely whether the game ends (task termination) and whether there is a single outcome of the game (which hopefully corresponds to successful completion in case the task is feasible).

A. Robot Part Mating

The mating control laws are defined by a collection of smooth scalar valued maps functions $\varphi_i : \mathbf{R}^2 \times \mathbf{R}^{2p} \to \mathbf{R}$, $\forall i \in P$. Each $\varphi_i(r, b)$ is defined as

$$\varphi_i(r, b) = \frac{\gamma_i^{k_2}(r, b)}{\beta_r(r, b)}$$

where $\gamma_i(r,b) : \mathbf{R}^2 \times \mathbf{R}^{2p} \to \mathbf{R}$ is the squared Euclidean distance between the robot and part i defined as $\gamma_i(r,b) = \|r - b_i\|^2, \forall i \in P$. The obstacle function $\beta_r(r,b) : \mathbf{R}^2 \in \mathbf{R}^{2p} \to \mathbf{R}$ is defined as $\beta_r(r,b) = \prod_{i\in P} \|r - b_i\|^2 - (\rho_r + \rho_i)^2, \forall i \in P$. The constant $k_2 \in \mathbf{Z}^+$ is a positive integer chosen appropriately.

The robot's motion toward part i is governed by the dynamical system

$$\dot{r} = w_i(r,b) \tag{1}$$

where the vector field w_i is defined by the negative gradient of φ_i as $w_i(r,b) = -D_r\varphi_i(r,b)$.

The integral curve of $\dot{r}$ through the initial condition $(r[m],\ b[m])$ will be defined by $w_i^t(r[m], b[m])$ [12]. If $w_i(r^*, b[m]) = 0$ implies that $D_r w_i(r^*, b[m])$ has full rank, then the limit set $\lim_{t\to\infty} w_i^t(r[m], b[m]) = w_i^\infty(r[m], b[m])$ through every initial condition $(r[m],\ b[m])$ is some isolated point $r[m+1] \triangleq w_i^\infty(r[m]; b[m])$—which corresponds to robot-part mating configuration. Otherwise, $r[m+1] \in w_i^\infty(r[m], b[m])$—which possibly means that robot gets stuck at a local minimum and cannot mate with the designated part.

B. Robot Part Moving

Once the robot mates with part i, the robot-part coupled structure moves as a single body in the extended space SE(2). The position vector b_i of part i is dependent on the augmented state vector r_a as follows: $b_i = r + d\left[\begin{smallmatrix}\cos\theta \\ \sin\theta\end{smallmatrix}\right]$ where d denotes the mating distance between the robot and the mated part. Hence, a new set of control laws for robot part moving are required.

These control laws are defined by again considering a collection of smooth maps of $\psi_i : \mathrm{SE}(2) \times \mathbf{R}^{2p-2} \to \mathbf{R}, \forall i \in P$. Let $\bar{b}_i \triangleq \{b_1, \ldots, b_{i-1}, b_{i+1} \ldots b_p\}$. The construction of $\psi_i(r_a, \bar{b}_i)$ is as follows:

$$\psi_i(r_a, \bar{b}_i) = \frac{\gamma^{k_3}(r_a, \bar{b}_i)}{\beta_i(r_a, \bar{b}_i)}$$

where $\gamma(r_a, \bar{b}_i) : \mathrm{SE}(2) \times \mathbf{R}^{2p-2} \to \mathbf{R}$ denotes the total squared Euclidean distances between the current configuration of the parts and the goal configuration defined as

$$\gamma(r_a, \bar{b}_i) = \left\| r + d\begin{bmatrix}\cos\theta \\ \sin\theta\end{bmatrix} - g_i \right\|^2 + \sum_{j\in P}^{j\neq i} \|b_j - g_j\|^2.$$

$\beta_i(r_a, \bar{b}_i) : \mathrm{SE}(2) \times \mathbf{R}^{2p-2} \to \mathbf{R}, \forall i \in P$, denotes the obstacle space implicitly as

$$\beta_i(r_a, \bar{b}_i) = \prod_{j\in P}^{j\neq i} \left\{ \left[\|r - b_j\|^2 - (\rho_r + \rho_j)^2\right] \times \left[\left\| r + d\begin{bmatrix}\cos\theta \\ \sin\theta\end{bmatrix} - b_j \right\|^2 - (\rho_i + \rho_j)^2 \right] \right\}.$$

The parameter $k_3 \in \mathbf{Z}^+$ is chosen to be a positive integer.

The motion of robot carrying part i is defined via constructing

$$\dot{r}_a = z_i(r_a, \bar{b}_i) \tag{2}$$

where the vector field z_i is defined as $z_i(r_a, \bar{b}_i) = -D_{r_a}\psi_i(r_a(t), \bar{b}_i(t))$.

The integral curve of $\dot{r}_a$ through the initial condition $(r_a[n], \bar{b}_i[n])$ is defined by $z_i^t(r_a[n], \bar{b}_i[n])$. If $z_i(r_a^*, \bar{b}_i[n]) = 0$ implies that $D_{r_a} z_i(r_a^*, \bar{b}_i[n])$ has full rank, then the limit set $\lim_{t\to\infty} z_i^t(r_a[n], \bar{b}_i[n]) = z_i^\infty(r_a[n], \bar{b}_i[n])$ through every initial condition $r_a[n],\ \bar{b}_i[n]$ is some isolated point $r_a[n+1] \triangleq z_i^\infty(r_a[n], \bar{b}_i[n])$ which corresponds to a robot successfully moving the part to its designated goal. Otherwise, $r_a[n+1] \in z_i^\infty(r_a[n], \bar{b}_i[n])$ which possibly means that the robot gives up moving part i to its goal position before reaching it. In turn, the navigation of part i is governed by the control law given, $v_i(r, b_i; \bar{b}_i) = -D_{b_i}\psi_i(r_a, \bar{b}_i)$. Using a chain rule on the right-hand side, we have

$$\begin{aligned} v_i(r, b_i; \bar{b}_i) &= D_{b_i} r_a \circ \left(-D_{r_a}\psi_i(r_a, \bar{b}_i)\right) \\ &= D_{b_i} r_a \circ z_i(r_a, \bar{b}_i). \end{aligned}$$

C. Implementation of the Game

The robot has to decide which is the best part to move next after it has dropped a part at a location. Once the next part is determined, it then selects the appropriate low-level controller (mating or moving) to apply.

Let $h : \mathbf{R}^{2p} \to P$ be an index valued function with the property

$$h(b) = \arg\max_{i\in P} \left\{ \left\| \left(I_2 \otimes e_i^T\right) D_b\phi(b) \right\| \right\} \tag{3}$$

that is, a function which picks out the component of b—whose direction of descent on $\phi(b)$ is the steepest. The function $\phi(b) : \mathbf{R}^{2p} \to \mathbf{R}$ is defined as $\phi(b) = (\gamma^{k_1}(b)/\beta(b))$. $\gamma(b) : \mathbf{R}^{2p} \to \mathbf{R}$ denotes the total squared Euclidean distances between the current configuration of the parts and the goal configuration, $\gamma(b) = \|b - g\|^2$, $\beta(b) : \mathbf{R}^{2p} \to \mathbf{R}$ denotes the obstacle function of the pairwise part positions, $\beta(b) = \prod_{i\in P} \prod_{j\in P}^{j>i} \|b_i - b_j\|^2 - (\rho_i + \rho_j)^2$ and $k_1 \in \mathbf{Z}^+$, is a positive integer.

Once the robot picks out which part i to move next, a partition $\mathcal{P}_i$, defined either on the robot's configuration space $\mathbf{R}^2$ or its mated extension SE(2), designates which control law to apply. Each partition has three cells—labeled $s \in \{\mathrm{next\_part},\ \mathrm{mate\_part},\ \mathrm{move\_part}\}$. The cells are determined by the critical sets C_{φ_i} and C_{ψ_i} of φ_i and ψ_i, respectively. Let $N_e(C_{\varphi_i})$ and $N_\varepsilon(C_{\psi_i})$ where $N_\varepsilon(.)$ denotes the union of ε-neighborhoods of the these critical points—where $\varepsilon \in \mathbf{R}^+$ is an arbitrarily small design parameter. Each partition $\mathcal{P}_i$ is then defined by

$$s = \begin{cases} \mathrm{next\_part} & r \in N_\varepsilon(C_{\varphi_i}) \text{ and } r \notin N_\varepsilon(b_i) \\ & \quad \text{or } r_a \in N_\varepsilon(C_{\psi_i}) \\ \mathrm{mate\_part} & r \notin N_\varepsilon(C_{\varphi_i}) \text{ and } r_a \notin N_\varepsilon(C_{\psi_i}) \\ \mathrm{move\_part} & r \in N_\varepsilon(C_{\varphi_i}) \text{ and } r_a \notin N_\varepsilon(C_{\psi_i}) \\ & \quad \text{and } r \in N_\varepsilon(b_i). \end{cases} \tag{4}$$

When a transition $s = \mathrm{next\_part}$ occurs, the switching mechanism is invoked and a new part $h(b)$ is selected and new partition $\mathcal{P}h(b)$ governs the control decisions. If during $\mathrm{mate\_part}$ stage, motion of the robot is blocked, the stage is aborted and a transition to the $\mathrm{next\_part}$ stage occurs. If the $\mathrm{mate\_part}$ stage ends up at a robot position close enough to the target part then the robot grasps the part and a transition to $\mathrm{move\_part}$ stage occurs. If during $\mathrm{move\_part}$ stage, motion of the robot and part pair is blocked, the stage is aborted, and a transition to the $\mathrm{next\_part}$ stage occurs. Hence, robot's motion is defined by the

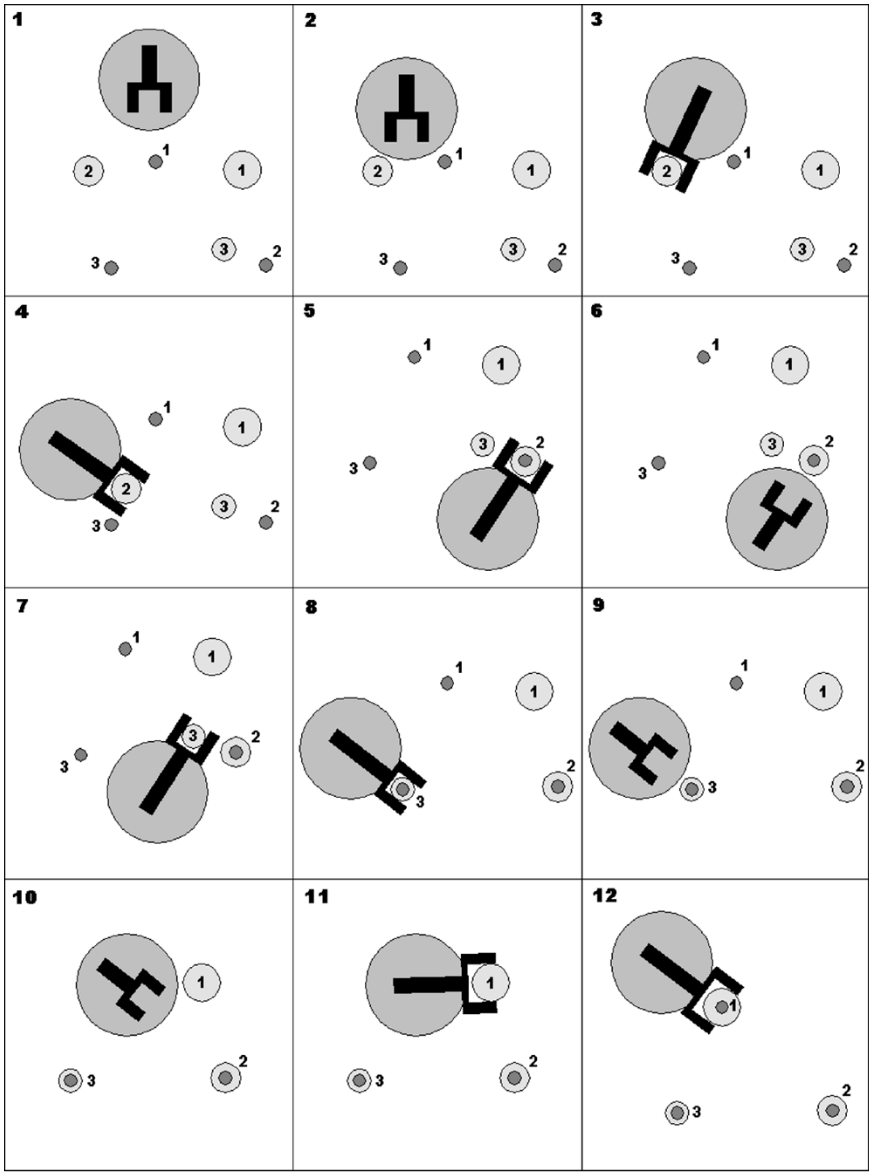

Fig. 3. Snapshots of an experiment with three parts: the big black circle represents EDAR and the numbered white circles represent the parts.

transition map from one blocked configuration $r_a[m] = [r[m]\ \ \theta[m]]^T$ and $b[m]$ to the other $r_a[m+1]$ as follows:

$$r_a[m+1] \in \begin{cases} \begin{bmatrix} w_h^{\infty}(r[m],b[m]) \\ \theta[m] \end{bmatrix}, & s = \mathrm{mate\_part} \\ z_h^{\infty}\left(r_a[m], \bar{b}_h[m]\right), & s = \mathrm{move\_part} \\ \left\{r_a[m]\right\}, & s = \mathrm{next\_part}. \end{cases} \tag{5}$$

The resulting dynamical system[2] can be interpreted as a noncooperative game played among the parts. The fixed points of the map governing the behavior of this system are the solutions of the game. Unlike the cooperative [19] and Nash game settings [5] that were respectively used for the exogeneous and endogeneous versions of the simplified one-dimensional (1-D) version of the problem, it is no longer clear what kind of equilibrium point is attained. Nash equilibria that prevail in the case of multiple, coupled and simultaneous objectives [4] may no longer be applicable due to the urgency ordering imposed by the sequential nature of the task.

[2]The iterated map of (5) over a state space indeed defines a dynamical system, albeit not the Lagrangian dynamics familiar to robot mechanical control problems. As it turns out, a gradient vector field does indeed lift to become a closed-loop controller for the Lagrangian dynamics [20], but that idea lies beyond the scope of the present paper.

III. EXPERIMENTAL RESULTS

EDAR[3] is a mobile robot with approximate translational velocity of 6 cm/s and an accompanying cumulative odometry error of roughly 2–3 cm after 100 cm of travel. The rotational inaccuracy percentage is roughly constant with a maximum value of six percent for a target 80^{circ} range. When its arm is at its nest position, its orthographic projection on to 2-D is a disk. The vision system provides an overhead orthographic view of the robot workspace and affords visual feedback regarding the position and the size of the parts and the robot as well as the angle of the robot gripper with respect to the robot base. With this system, EDAR can request visual feedback at the end of each $\mathrm{next\_part}$, $\mathrm{mate\_part}$, and $\mathrm{move\_part}$ states.[4] Between state changes, EDAR uses feedback from its optical encoders to update both its position and those of the parts. A typical run of an experiment with three parts is presented in Fig. 3.

[3]A detailed technical specification is provided in [16].

[4]It is certainly possible to endow EDAR with a more sophisticated and real-time vision module for recognizing and locating the parts as we plan to do in the future. However, this independent research topic lies well beyond the scope of the research reported here related to parts moving, for which the current vision affords an adequate experimental setting.

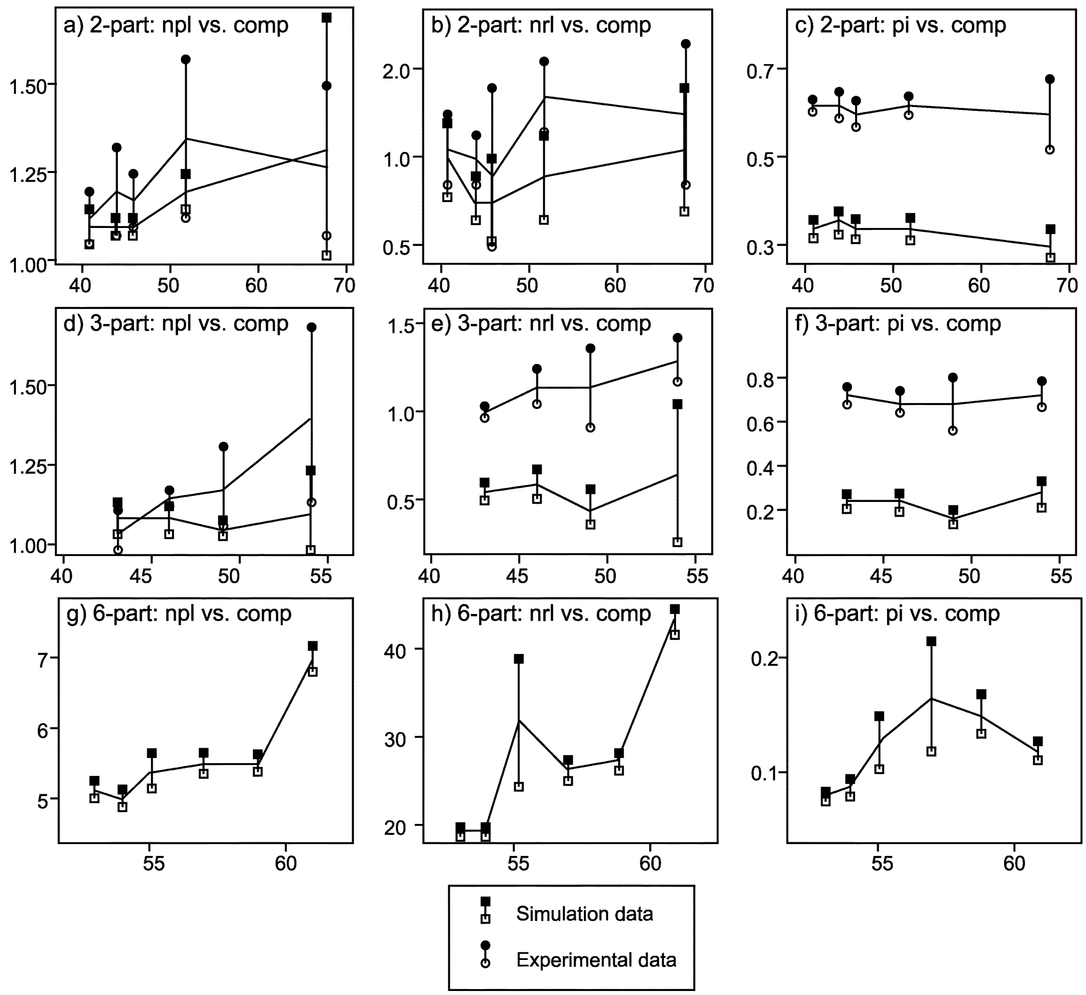

Fig. 4. Performance graphics. Each data point represents the mean and the standard deviation of 8–10 sample runs with random initial configurations. Note that the ordinate scales are presented in terms of dimensionless measures defined in the text according to the label of each figure.

A. Task Measures

We focus on the impact of task "difficulty"[5] by increasing either the number of movable obtacles, measured by $\mathrm{comp} = (100\binom{p}{2}/\log\beta)$, or the required "tightness" of the packing as measured by the scalar function $\beta = \prod_{(i,j)\in Q}[\|g_i - g_j\|^2 - (\rho_i + \rho_j)^2]$. The performance measures are defined as:

Normalized part path length (npl): The average of the total distances traveled by the parts from their initial positions to their final positions normalized by the Euclidean distance between the initial and goal configurations:

$$\mathrm{npl} = \frac{1}{p}\sum_{i\in P}\frac{\int_0^{t_f}\left\|\dot{b}_i(t)\right\|dt}{\|b_i(0) - g_i\|}$$

where t_f denotes the duration of a task. Note that, in general, npl has to be greater than 1 and the normalization is introduced in order to account for variations in the initial conditions.

Normalized robot path length (nrl): The total distance traveled by the robot from its initial position to the final position normalized by the sum of the Euclidean distance between the initial position of the robot and the parts and the Euclidean distances between the pairwise part goal positions

$$\mathrm{nrl} = \frac{\int_0^{t_f}\|\dot{r}(t)\|\,dt}{\sum_{i\in P}\|r(0) - b_i(0)\| - (\rho_i + \rho_r) + \sum_{(i,j)\in Q}\|g_i - g_j\|}$$

where $r(0)$ denotes the initial position of the robot.

Positional inaccuracy (pi): The average error based on the Euclidean distance between the goal position of the parts and their actual final positions normalized by the parts' radii

$$\mathrm{pi} = \frac{1}{p}\sum_{i\in P}\frac{1}{\rho_i}\|b_i(t_f) - g_i\|$$

where $b_i(t_f)$ denotes the final position of part i.

B. Simulations and Experiments

EDAR's workspace is restricted to a 2.5 m $\times$ 2.5 m area. As it needs to move away from obstacles, with this limited space, experiments involving at most three parts can be performed. Plastic cylindrical objects varying 6–11 cm in radius and 1.5 m in height are used as parts. For each set of experiments, five different randomly chosen goal configurations with increasing complexity are used. EDAR completes a twopart moving task in 5–10 min depending on the workspace complexity when $k_1 = k_2 = k_3 = 18$. The parameter k_i, $i = 1$, 2, and 3 values are chosen to optimize the tradeoff between space (longer path, more sloppy motion) and time (shorter path, higher accuracy of motion). The graphs shown in Fig. 4 present simulation (two, three, and six parts)

[5]Other measures such as p-handedness, nature and number of degrees of freedom required, and length of the longest sequence of instructions that have been proposed for open-loop approaches [31] cannot be computed *a priori* in feedback approaches.

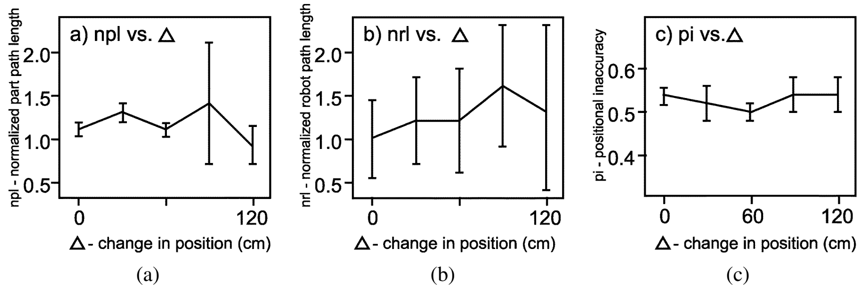

Fig. 5. Statistics of changing two parts' position experiments. The total amount of these perturbations—Δ is 30, 60, 90 and 120 cm. (a) Normalized part path length. (b) Normalized robot path length. (c) Positional inaccuracy.

and experiment (two and three parts only) performance measures. It is not surprising that npl and nrl values increase with difficulty, but interestingly, none exhibit any exponential trends—neither respecting increased number of parts nor tightness—as might be imagined given the complexity of the general problem. Note that this is a qualitative impression rather than a formal observation. As expected, the experimental values are of higher magnitude compared to their simulated counterparts due to robot and vision inaccuracies. Finally, increasing the k_i parameters that need to accompany the increasing complexity does not have much real impact on the positioning inaccuracy. We observe that the final accuracy of each placement at the "coarse" scale achieved would need, in a practical implementation, to be remedied by recourse to more specialized local adjunct planners whose consideration lies beyond the scope of this paper. These results are very encouraging as they suggest that the algorithm, free of any "re-planning" requirement, does not seem to incur the heavy computational burden introduced by the inevitable iteration of open-loop planning.

The highlight of our experiments, the central evidence that most corroborates the value of event-driven reactive planning, is obtained from the robot's response to unanticipated perturbations in the parts' positions. In these experiments, two parts are used and the goal configuration with a workspace complexity of $\mathrm{comp} = 46$ is chosen. While EDAR performs the moving task, the positions of the parts are changed randomly. As EDAR obtains visual feedback at the end of $\mathrm{mate\_part}$, $\mathrm{move\_part}$, $\mathrm{grasp\_part}$, or $\mathrm{ungrasp\_part}$, only relative changes at these instances can be feedback from the vision system. The reader may note that this represents a limitation of the visual feedback system rather than a limitation of the approach. Fig. 5 shows the performance measures as a function of Δ—which are similar to the unperturbed case.[6]

IV. Conclusion

This paper explores a feedback-based event-driven version of the warehouseman's problem wherein the actuated machine inhabits the same workspace as a collection of unactuated but movable parts. The problem is formulated as a noncooperative game—introducing artificial potential functions as "costs" to be interpreted at runtime as motion generation primitives. An extensive and systematic experimental assessment uses EDAR—the first physical implementation of purely feedback-based event-driven parts moving by a mobile robot. The empirical near real-time performance and robustness against (admittedly limited) workspace changes motivates our ongoing research efforts toward an associated formal analysis of convergence (i.e., successful task completion). Interestingly, to the best of the authors' knowledge, even for the open-loop version of the problem, there remains no general provably complete result. Obviously, the present setting needs to be generalized to more complex and realistic scenarios before it can be applied in a real implementation.

[6]It should be noted that the time required changes only as a function of the perturbation amount. This is expected since the paths taken by the parts may decrease or increase as an amount of Δ depending on their new positions.

Acknowledgment

The authors would like to thank the editor and the anonymous reviewers for their constructive criticism of the earlier version of the paper.

References

[1] T. Basar and G. J. Olsder, *Dynamic Noncooperative Game Theory*. New York: Academic, 1982.

[2] O. Ben-Shahar and E. Rivlin, "To push or not to push: On the rearrangement of movable objects by a mobile robot," *IEEE Trans. Syst., Man, Cybern. B*, vol. 28, pp. 667–679, Oct. 1998.

[3] ——, "Practical pushing planning for rearrangement tasks," *IEEE Trans. Robot. Automat.*, vol. 14, pp. 549–565, Aug. 1998.

[4] H. I. Bozma and J. S. Duncan, "A game-theoretic approach to integration of modules," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 16, pp. 1074–1086, Nov. 1994.

[5] H. I. Bozma and D. E. Koditschek, "Assembly as a noncooperative game of its pieces: analysis of 1D sphere analysis," *Robotica*, vol. 19, pp. 93–108, 2001.

[6] R. Chatila, "Deliberation and reactivity in autonomous mobile robots," *Robot. Auton. Syst.*, vol. 16, pp. 197–211, 1995.

[7] J. Chuang, "Potential-based modeling of three-dimensional workspace for obstacle avoidance," *IEEE Trans. Robot. Automat.*, vol. 14, pp. 778–785, May 1998.

[8] B. Dacre Wright, J. P. Laumond, and R. Alami, "Motion planning for a robot and a movable object amidst polygonal obstacles," in *Proc. 1992 IEEE Int. Conf. Robotics and Automation*, Nice, France, 1992, pp. 2474–2480.

[9] J. M. Evans, "HelpMate: An autonomous mobile robot courier for hospitals," in *Proc. Int. Conf. Intelligent Robots and Systems*, Munich, Germany, 1994, pp. 1695–1700.

[10] S. S. Ge and Y. J. Cui, "New potential functions for mobile robot path planning," *IEEE Trans. Robot. Automat.*, vol. 16, pp. 615–620, Oct. 2000.

[11] R. Goodwin, K. Z. Haigh, S. Koenig, and J. O'Sullivan, "A layered architecture for office delivery robots," *Auton. Agents*, pp. 235–242, 1997.

[12] J. Guckenheimer and P. Holmes, *Nonlinear Oscillations, Dynamical Systems and Bifurcations of Vector Fields*. New York: Springer-Verlag, 1983, p. 43.

[13] D. Halperin, L. Kavraki, and J.-C. Latombe, *Handbook of Discrete and Computational Geometry*, J. E. Goodman and J. O'Rourke, Eds. Boca Raton, FL: CRC Press, 1997, pp. 755–778.

[14] Y. K. Hwang and N. Ahuja, "Gross motion planning—A survey," *ACM Comput. Surveys*, vol. 24, no. 3, pp. 219–291, 1992.

[15] J. E. Hopcroft, J. T. Schwartz, and M. Sharir, "On the complexity of motion planning for multiple independent objects; PSPACE-hardness of the warehouseman's problem," *Int. J. Robot. Res.*, vol. 3, no. 4, pp. 76–88, 1984.

[16] C. S. Karagöz, "A game-theoretic approach to objects' moving problem with mobile robots," Ph.D. dissertation, Elect. Electron. Eng. Dept., Bogazici Univ., Istanbul, Turkey, 2001.

[17] C. S. Karagöz, H. I. Bozma, and D. E. Koditschek, "Coordinated motion of disk-shaped independent robots in 2D workspaces," Univ. Michigan, Ann Arbor, Tech. Rep. CSE-TR-486-04, Feb. 2004.

[18] O. Khatib, "Real time obstacle avoidance for manipulators and mobile robots," *Int. J. Robot. Res.*, vol. 5, no. 1, pp. 90–99, 1986.

[19] D. E. Koditschek, "An approach to autonomous robot assembly," *Robotica*, vol. 12, no. 2, pp. 137–155, 1994.

[20] ——, *The Application of Total Energy as a Lyapunov Function for Mechanical Control Systems, in Control Theory and Multibody Systems*, J. Marsden, P. S. Krishnaprasad, and J. Simo, Eds. Providence, RI: American Mathematics Society, 1989, pp. 131–158.

[21] J.-C. Latombe, *Robot Motion Planning*. Boston, MA: Kluwer, 1991.

[22] S. M. LaValle and S. Hutchinson, "Optimal motion planning for multiple robots having independent goals," *IEEE Trans. Robot. Automat.*, vol. 14, pp. 912–925, Dec. 1998.
[23] T. Lozano-Perez, J. L. Jones, E. Mazer, P. A. O'Donell, and W. E. L. Grimson, "Handey: A robot system that recognizes, plans and manipulates," in *Proc. IEEE Int. Conf. Robotics and Automation*, 1987, pp. 843–849.
[24] W. S. Newman and N. Hogan, "High speed robot control and obstacle avoidance using dynamic potential functions," in *Proc. IEEE Conf. Robotics and Automation*, 1987, pp. 14–24.
[25] J. Ota, "Rearrangement planning of multiple movable objects by using realtime search methodology," in *Proc. 2002 IEEE Int. Conf. Robotics and Automation*, Washington, DC, 2002, pp. 947–953.
[26] J. Reif and M. Sharir, "Motion planning in the presence of moving obstacles," *J. ACM*, vol. 41, no. 4, pp. 764–790, 1994.
[27] E. Rimon and D. E. Koditschek, "Exact robot navigation using artificial potential functions," *IEEE Trans. Robot. Automat.*, vol. 8, pp. 501–518, Oct. 1992.
[28] M. A. Salichs and L. Moreno, "Navigation of mobile robots: Open questions," *Robotica*, vol. 18, pp. 227–234, 2000.
[29] M. Sharir, *Handbook of Discrete and Computational Geometry*, J. E. Goodman and J. O'Rourke, Eds. Boca Raton, FL: CRC Press, 1997, pp. 733–754.
[30] C. Warren, "Global path planning using artificial potential fields," in *Proc. IEEE Conf. Robotics and Automation*, 1989, pp. 321–336.
[31] R. H. Wilson and J.-C. Latombe, "Geometric reasoning about mechanical assembly," *Artif. Intell.*, vol. 71, pp. 371–396, 1994.
[32] G. Wilfong, "Motion planning in the presence of movable obstacles," in *Proc. 4th Annu. ACM Symp. Computational Geometry*, 1988, pp. 279–288.

Path Following by the End-Effector of a Redundant Manipulator Operating in a Dynamic Environment

Mirosław Galicki

***Abstract*—This paper addresses the problem of generating at the control-loop level a collision-free trajectory for a redundant manipulator operating in dynamic environments which include moving obstacles. The task of the robot is to follow, by the end-effector, a prescribed geometric path given in the work space. The control constraints resulting from the physical abilities of robot actuators are also taken into account during the robot movement. Provided that a solution to the aforementioned robot task exists, the Lyapunov stability theory is used to derive the control scheme. The numerical simulation results for a planar manipulator whose end-effector follows a prescribed geometric path, given in both an obstacle-free work space and a work space including the moving obstacles, illustrate the trajectory performance of the proposed control scheme.**

***Index Terms*—Collision-free trajectory, dynamic environment, Lyapunov stability, redundant manipulator.**

I. Introduction

In recent years, interest has increased in applying redundant manipulators in useful practical tasks which are specified in terms of a geometric path to be followed by the end-effector. Application of redundant manipulators to such tasks complicates their performance, since these manipulators, in general, do not provide unique solutions. Consequently, some objective criteria should be specified to solve the robot tasks uniquely. Minimization of the performance time is mostly considered in the literature. One may distinguish several approaches in this context. The task of time-optimal control of nonredundant manipulators has been solved by means of effective algorithms in [1]–[8]. Using the concept of a regular trajectory and the extended state space, numerical procedures were proposed in [9]–[12] to find path-constrained time-optimal controls for kinematically redundant manipulators. Although all of the aforementioned algorithms produce optimal solutions, they are not suitable for real-time computations due to their computational complexity. Therefore, it is natural to attempt other techniques in order to generate the robot trajectory in real time. Using an integrable manifold concept and the inverse of the extended Jacobian matrix, an algorithm has been proposed in [13] to determine robot motions satisfying the end-effector path constraints and other useful objectives. A kinematic singular path-tracing approach (based on the null-space-based method [14]) has been presented in [15]. Recently, a technique which avoids solving an inverse of robot kinematic equations has been offered in [16] for determining a collision-free trajectory of redundant manipulators operating in a static environment.

This paper is a generalization of the results obtained in [16]. Namely, it presents an approach to the problem of controlling a redundant manipulator so that its end-effector follows a prescribed geometric path, and the manipulator simultaneously avoids collisions with moving obstacles in the work space. In addition, the constraints imposed on the robot controls are taken into account. Provided that a solution to the control problem of redundant manipulators exists, the Lyapunov stability theory is used to derive the trajectory generator. The approach offered does not require any inverse of robot kinematic equations. Instead, a transpose Jacobian matrix is used to generate robot motions. The idea of applying the transpose Jacobian technique to solving the inverse kinematic problem for both nonredundant and redundant manipulators has been extensively studied in [17]–[20] and generalized in [21]–[25] to the case where joint velocities [21] and accelerations constraints [22]–[25] are taken into account. Based on a slowing-down technique, the work in [21]–[25] (locally) rescales the desired (planned) end-effector trajectory when joint limits are encountered. This paper presents an alternative approach to the inverse kinematic problem discussed in [22]–[25]. The time evolutions of both manipulator configuration and path parameter are derived here from the Lyapunov stability theory, whereas the work in [21]–[25] solved a one-dimensional optimization problem in each time interval to find a time warp when joint limits are encountered. The gain coefficients in both algorithms are determined based on quite different kinds of information, which makes the performance of these algorithms, in fact, numerically incomparable. This is due to the fact that control gains proposed herein are functions of torque limits, whereas the time warp from [22]–[25] depends on either velocity or acceleration constraints. An alternative method adding constraints (e.g., collision–avoidance constraints) to solve the manipulator redundancy is the configuration control method proposed in [26] and [27], which, however, introduces additional singularities related to a user-specified task. It also seems difficult to use this approach particularly in situations where the number of active collision–avoidance constraints exceeds the degree of manipulator redundancy. Moreover, the existing configuration control schemes are, in fact, suitable for the end-effector trajectory tracking, and not for the path following considered herein.

Furthermore, it is also shown here how, through the use of the exterior penalty function method [28], the collision–avoidance constraints

Manuscript received March 3, 2003; revised October 16, 2003 and January 7, 2004. This paper was recommended by Associate Editor S. Chiaverini and Editor A. De Luca upon evaluation of the reviewers' comments. This work was supported by DFG Ga 652/1-1.

The author is with the Institute of Medical Statistics, Computer Science and Documentation, Friedrich Schiller University, D-07740 Jena, Germany, and also with the Department of Management, University of Zielona Góra, 65-246 Zielona Góra, Poland (e-mail: galicki@imsid.uni-jena.de).

Digital Object Identifier 10.1109/TRO.2004.833782