University of Pennsylvania
ScholarlyCommons

Departmental Papers (ESE) | Department of Electrical & Systems Engineering

February 1990

A Family of Robot Control Strategies for Intermittent Dynamical Environments

Martin Buehler
Yale University

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

Peter J. Kindlmann
Yale University

Follow this and additional works at: https://repository.upenn.edu/ese_papers

Recommended Citation

Martin Buehler, Daniel E. Koditschek, and Peter J. Kindlmann, "A Family of Robot Control Strategies for Intermittent Dynamical Environments", . February 1990.

Copyright 1990 IEEE. Reprinted from *IEEE Control Systems Magazine*, Volume 10, Issue 2, February 1990, pages 16-22.

This material is posted here with permission of the IEEE. Such permission of the IEEE does not in any way imply IEEE endorsement of any of the University of Pennsylvania's products or services. Internal or personal use of this material is permitted. However, permission to reprint/republish this material for advertising or promotional purposes or for creating new collective works for resale or redistribution must be obtained from the IEEE by writing to pubs-permissions@ieee.org. By choosing to view this document, you agree to all provisions of the copyright laws protecting it.

NOTE: At the time of publication, the author Daniel Koditschek was affiliated with Yale University. Currently, he is a faculty member of the School of Engineering at the University of Pennsylvania.

This paper is posted at ScholarlyCommons. https://repository.upenn.edu/ese_papers/336
For more information, please contact repository@pobox.upenn.edu.

A Family of Robot Control Strategies for Intermittent Dynamical Environments

Abstract

This article develops a formalism for describing and analyzing a very simple representative class of robotic tasks that require "dynamical dexterity" - among them, the task of juggling. The authors review their empirical success, to date, with a new class of control algorithms for this task domain, called "mirror algorithms." The formalism for representing the task domain and encoding within it the desired robot behavior enables them to prove that a suitable mirror algorithm is correct with respect to a specified task.

Comments

Copyright 1990 IEEE. Reprinted from *IEEE Control Systems Magazine*, Volume 10, Issue 2, February 1990, pages 16-22.

This material is posted here with permission of the IEEE. Such permission of the IEEE does not in any way imply IEEE endorsement of any of the University of Pennsylvania's products or services. Internal or personal use of this material is permitted. However, permission to reprint/republish this material for advertising or promotional purposes or for creating new collective works for resale or redistribution must be obtained from the IEEE by writing to pubs-permissions@ieee.org. By choosing to view this document, you agree to all provisions of the copyright laws protecting it.

NOTE: At the time of publication, the author Daniel Koditschek was affiliated with Yale University. Currently, he is a faculty member of the School of Engineering at the University of Pennsylvania.

This journal article is available at ScholarlyCommons: https://repository.upenn.edu/ese_papers/336

A Family of Robot Control Strategies for Intermittent Dynamical Environments

Martin Bühler, Dan E. Koditschek, and Peter J. Kindlmann

ABSTRACT: This article develops a formalism for describing and analyzing a very simple representative class of robotic tasks that require "dynamical dexterity"—among them, the task of juggling. The authors review their empirical success, to date, with a new class of control algorithms for this task domain, called "mirror algorithms." The formalism for representing the task domain and encoding within it the desired robot behavior enables them to prove that a suitable mirror algorithm is correct with respect to a specified task.

Introduction

We are interested in robotic task domains involving intermittent dynamical environments, and this article considers a simple representative from a range of robotic tasks associated with dexterous capabilities that might be grouped under the general rubric of "juggling." This term includes those tasks requiring throwing and catching, or (as in this article) beating and batting, or any other interaction with an object (or multiple objects), which would otherwise fall freely in the earth's gravitational field. Such tasks share the property of presenting nontrivial dynamical environments whose characteristics change intermittently subject to excitation from the robot. It seems fair to say that the only systematic work in this realm to date has been the pioneering research of Raibert, whose careful experimental studies verify the correctness of his elegant control scheme [1]. Our strategy for research in this area is to study, carefully, a very simple experimental apparatus and develop a theoretical perspective that both explains the particular empirical experience and generalizes to a synthesis procedure over the larger task domain. This article reviews our own experimental results and presents a body of theory that accomplishes the first goal.

Presented at the 1989 IEEE International Conference on Robotics and Automation, Scottsdale, Arizona, May 15-19, 1989. The authors are with the Center for Systems Science, Department of Electrical Engineering, Yale University, New Haven, CT 06520.

The experimental setup and a simplified mathematical model are presented in the next section. Analysis of the "contact geometry" between robot and environment gives rise to an impact model—the "environmental control system"—with respect to which the "vertical one-juggle" task is formally defined and proven to be achievable in the following section. Next, we offer a review of previous results, introduce a family of robot control strategies arising from a "mirror geometry" in the phase space of the robot-environment pair, demonstrate that this family solves the environmental control problem—the rigorous formulation of the juggling task expressed in terms of the contact geometry—and present experimental data attesting to the physical validity of this strategy. Essentially, the controller has been designed so that the desired juggling pattern is an attracting periodic orbit of the closed-loop robot-environment dynamics. The conclusion offers some speculations upon the larger implications of this work for robotic tasks in more general intermittent dynamical environments.

The Empirical and Analytical Setting

This section introduces the experimental apparatus—the Yale Planar Juggling Robot—and develops a simplified mathematical model of the physics relevant to the juggling task.

Mechanical and Computational Setup

The physical apparatus consists of a puck, which slides on an inclined plane and is batted successively by a simple "robot": a bar with a billiard cushion rotating in the juggling plane, as depicted in Fig. 1.

All intelligent sensor and controller functions are performed by a four-node distributed computational network formed from Yale XP/DCS control nodes [2]. In addition to the 10 MIPS, 1.5 MFLOPS, and high-speed (20, 10, or 5 MBits/sec) serial interprocessor communications rate contributed by the INMOS Transputer, our XP/DCS boardset features fiber optics support, fast RAM, as well as support for fast and extensive I/O via a bidirectional latched 32-bit I/O bus.

In order to move the bar according to some puck-dependent control algorithm, the puck's position and velocity must be measured. Presently, this is accomplished by placing an oscillator inside the puck and burying a grid in the juggling plane, thus imitating a big digitizing tablet. The sensor node measures induced voltages and computes the puck positions from the zero- and first-order moments. These, in turn, feed into a standard linear observer to reduce measurement noise in position and estimate velocity. The output of the observer is communicated asynchronously via fiber optics to the juggle planning and control node at a rate of 1 kHz. There, a reference trajectory for the motor is computed, according to the mirror algorithm that we describe below, and communicated to the motor controller node. The latter issues torque commands to the motor at a rate of 2 kHz, performing noise filtering and numerous additional housekeeping and safety checks as well. The fourth node is used as a logging system and human interface. At the time of this writing, we are completing a real-time vision system [3] based on the XP/DCS in order to move off the plane and juggle in three spaces.

A Simplified Mathematical Model

Locate a frame of reference, $\mathfrak{F}_0$, at the center of the robot shaft, with the b_1-axis and b_2-axis as depicted in Fig. 2. Define q so that it measures the angle of the right-hand portion of the robot's bar (with the hitting surface—the billiard cushion—facing up) away from the b_1-axis on the juggling plane.

The Robot and Environment Models The configuration space of the entire problem is the cross product, $\mathcal{C} \triangleq \mathcal{B} \times \mathcal{Q}$, of the environment and the robot configurations. We will model the robot's configuration space as $\mathcal{Q} \triangleq [-\pi/2, \pi/2]$, real scalars, which we restrict to a half-revolution, since, for present purposes, it will suffice to consider only those locations of the bar in the right half of the juggling plane for which the hitting bil-

0272-1708/90/0200-0016 $01.00 © 1990 IEEE

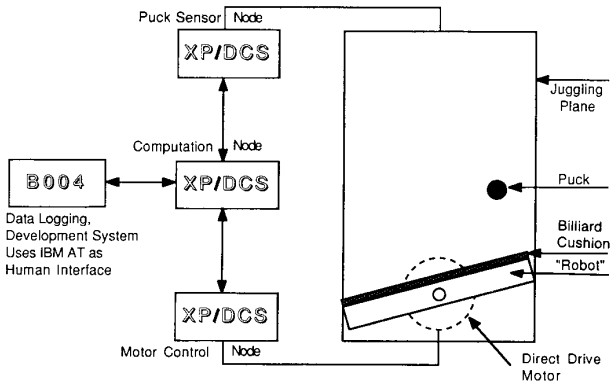


Fig. 1. The Yale Juggler.

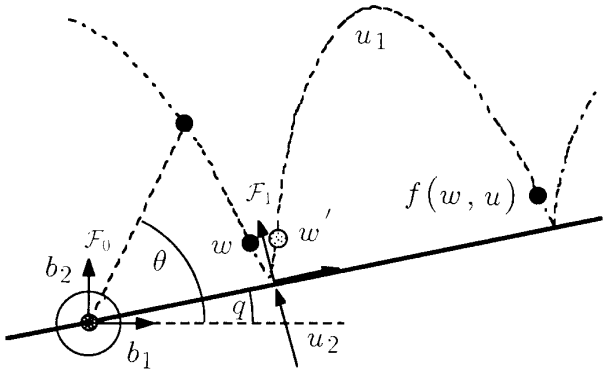


Fig. 2. The impact event.

liard cushion is facing up. We will represent the location of the falling body on the plane as a real two vector in $\mathcal{B}$ with the coordinates (b_1, b_2) denoting, respectively, the position of its centroid relative to the "horizontal" (b_1) and "vertical" (b_2) axes of the reference frame, $\mathcal{F}_0$.

In isolation, the robot's dynamics occur in its phase space, $\mathcal{R} \triangleq T\mathcal{Q} \approx \mathcal{Q} \times \mathbb{R}$, of angular positions and velocities and may be modeled simply by the following equations (where v denotes the commanded torque from the motor control node and ρ denotes the moment of inertia of the bar) since the motor (with its high bandwidth and power, low shaft friction, and inertia deployed in the absence of any transmission) comes close to providing a source of "pure torque."

$$\begin{bmatrix} \dot{r}_1 \\ \dot{r}_2 \end{bmatrix} \triangleq \begin{bmatrix} \dot{q} \\ \ddot{q} \end{bmatrix} = \begin{bmatrix} r_2 \\ v/\rho \end{bmatrix} \tag{1}$$

Unfortunately, the large mass of this motor mitigates against its role in a multijointed direct drive robot.

In isolation, the puck's dynamics occur in its phase space, $\mathcal{W} \triangleq T\mathcal{B} \approx \mathcal{B} \times \mathbb{R}^2$, and may be modeled by the following equations (where $a = [0, -\gamma]^T$) since we assume that the puck is a point of unit mass sliding on a frictionless surface.

$$\begin{bmatrix} \dot{w}_1 \\ \dot{w}_2 \end{bmatrix} \triangleq \begin{bmatrix} \dot{b} \\ \ddot{b} \end{bmatrix} = \begin{bmatrix} w_2 \\ a \end{bmatrix} \triangleq n(w) \tag{2}$$

In fact, this idealized model is overly simplistic, since there is noticeable Coulomb friction on the sliding plane. One of the objectives of our study is to develop a control procedure that is robust enough to succeed even in the face of such unmodeled dynamics, and we will use only n from (2) in the formal analysis. However, in the sequel, we will find it interesting to compare numerical simulations of the robot control laws in the idealized environment, n, with the same strategies run in the more realistic simulation model with friction, as against empirical data.

Finally, the set of all possible impact configurations may be described by a smooth surface, $\mathcal{I}$, in the puck-robot configuration space, $\mathcal{C}$, as formalized in [4, Lemma 2.1]. An impact configuration, $(q, b) \in \mathcal{I}$, implicitly defines the robot's "virtual gripper"—the point of contact on the billiard cushion; it is useful to define a new "virtual gripper frame," $\mathcal{F}_1$, whose origin is in the body's center, b, and whose b_1 and b_2 axis are aligned with, and perpendicular to, the robot bar, respectively—all depicted in Fig. 2. The new frame has a representation with respect to the "base frame" given by

$${}^0\mathcal{F}_1 = \begin{bmatrix} R & b \\ 0^T & 1 \end{bmatrix}$$

$$R \triangleq \begin{bmatrix} \cos q & -\sin q \\ \sin q & \cos q \end{bmatrix} \tag{3}$$

The Impact Model We now develop a simplified model of the dynamics of repeated puck-robot impacts, based upon the following assumptions. First, we assume here that all interactions between the ball and robot during impact can be adequately modeled as an instantaneous event: a posteriori velocities are related to a priori velocities via a simple "coefficient of restitution" [5], α, which belongs to the interval (0, 1). Second, we assume that the robot mass is sufficiently large as to make the puck's mass negligible. Finally, we neglect puck spin, and it is assumed that the puck's velocity component parallel to the robot bar is unchanged by the impact.

Under these assumptions, the a posteriori velocity of the body after impact, $\dot{b}'$, is related to the a priori velocity of the body, $\dot{b}$, and that of the robot's virtual gripper, $u_2 \triangleq \|b\| \cdot \dot{q}$, in the $\mathcal{F}_1$ coordinates as

$${}^1\dot{b}' = \begin{bmatrix} 1 & 0 \\ 0 & -\alpha \end{bmatrix} {}^1\dot{b} + \begin{bmatrix} 0 \\ 1 + \alpha \end{bmatrix} u_2$$

$$= \bar{C}\,{}^1\dot{b} + \bar{c}u_2$$

This is expressed in $\mathcal{F}_0$ coordinates as

$$\dot{b}' = C\dot{b} + cu_2 \tag{4}$$

where

$$C \triangleq R\bar{C}R^T$$

$$c \triangleq R\bar{c}$$

Recall that R and, hence, C, c are all functions of q. But since $q = \theta(b)$ for all impact configurations, $(b, q) \in \mathcal{I}$, we obtain $\dot{b}'$ purely as a function of $(b, \dot{b})$ and $\dot{q}$.

The forward trajectory of the body is now obtained by integrating its motion in $\mathcal{W}$ starting from the initial conditions, $w = (b, \dot{b}')$, according to the isolated dynamics, $n(w)$, given in (2),

$$w(t) = \begin{bmatrix} b + \dot{b}'t + \frac{1}{2}at^2 \\ \dot{b}' + at \end{bmatrix} \tag{5}$$

The Environmental Control Problem

This section investigates the response of the puck to all logically possible impact events by examining the *environmental control system*. This results from considering the effect of repeated puck-robot impacts on the future puck trajectory (5), assuming arbitrarily assigned values for the robot, $(q, \dot{q}) \in \mathcal{R}$, at each impact event, independent of the robot's dynamics (1). From (5), it is clear that the first time, t_{j+1}, after the jth impact at time t_j, at which the robot and body again make contact, $(b(t_{j+1}), q(t_{j+1})) \in \mathcal{I}$, is a function of the robot's future position trajectory, $q(t_{j+1})$. Moreover, from (4), it is clear that the velocity of the virtual gripper at impact is determined by choice of the robot's velocity at impact, $\dot{q}(t_j)$. In the sequel, we will use the term *impact schedule* to denote a sequence of pairs, where $u_2(t_j)$ denotes the velocity of the virtual gripper at the moment of the jth impact, which occurs at time $t = t_j$, and $u_1(t_j) \triangleq t_{j+1} - t_j$ denotes the interval of time that elapsed between that impact event and its successor.

$$\{u(t_j)\}_{j=0}^{\infty}$$

$$u(t_j) = \begin{bmatrix} u_1(t_j) \\ u_2(t_j) \end{bmatrix} \in \mathcal{U} \triangleq \mathbb{R}^2$$

An impact schedule gives rise to a sequence of puck states measured just before impact,

$$\{w(t_j)\}_{j=0}^{\infty}$$

$$w(t_{j+1}) = f(w(t_j), u(t_j))$$

The function $f: \mathcal{W} \times \mathcal{U} \to \mathcal{W}$ is derived by substituting (4) into (5) to obtain

$$f(w, u) = \begin{bmatrix} b + (C(b)\dot{b} + c(b)u_2)u_1 + \frac{1}{2}au_1^2 \\ C(b)\dot{b} + c(b)u_2 + au_1 \end{bmatrix} \tag{6}$$

This nonlinear discrete dynamical control system comprises the environmental control system. An *environmental control problem* results from prescribing some desired sequence of puck states, $\{w^*(t_j)\}_{j=0}^{\infty}$, and asking for an impact sequence, $\{u^*(t_j)\}_{j=0}^{\infty}$, which results in asymptotic convergence of $w(t_j)$ to $w^*(t_j)$.

Clearly, any control problem may be solved by a great variety of controller structures. This section solely concerns solutions via pure feedback compensation: namely, we shall abstract away all physical properties of the robot and presume it to be an "ideal" feedback agent that measures puck states, $w(t_j)$, and delivers control inputs, $u(t_j)$, accordingly. This point of view affords a precise definition of the juggling task in the next section, as well as the demonstration that the task is at least logically achievable in the following section.

The Vertical One-Juggle

Probably the simplest systematic behavior of this environment imaginable (after the rest position) is a periodic vertical motion of the puck in its plane. Specifically, one would like to be able to specify an arbitrary "apex" point in the juggling plane and, from arbitrary initial puck conditions, force the puck to attain a purely vertical periodic trajectory with the specified apex point.

Given the specific juggling task at hand, only a subset of the full puck impact phase space is of interest. As mentioned before, we are limiting the set points to the right-hand side $b_1 > 0$ of the juggling plane. Also, it only makes sense to admit negative vertical puck velocities $\dot{b}_2 < 0$ just before impact—i.e., those that point toward the robot's hitting bar. Thus, we define the *working* puck phase space $\tilde{\mathcal{W}} \subset \mathcal{W}$ as

$$\tilde{\mathcal{W}} \triangleq \{w \in \mathcal{W}: b_1 > 0, \dot{b}_2 < 0\}$$

Since a purely vertical trajectory requires zero horizontal velocity, $\dot{b}_1 = 0$, and a fixed vertical impact velocity, $\dot{b}_2^*$, from a specified impact height, and $b_2 = 0$ implies a specified apex position, one is led to the following definition. Let the task subspace of the vertical one-juggle be the plane

$$\mathcal{J} \triangleq \{w \in \tilde{\mathcal{W}}: b_2 = 0, \dot{b}_1 = 0\}$$

Say that a feedback law, $g: \mathcal{W} \to \mathcal{U}$, constitutes a *vertical one-juggle* with respect to the task, $w^* \in \mathcal{J}$, if w^* is a fixed point of the closed-loop system,

$$w^* = f_g(w^*)$$

$$f_g(w) \triangleq f(w, g(w)) \tag{7}$$

and is a stable attractor of the resulting discrete dynamics.

Proposition 1 ([4]) *Given the discrete dynamical control system (6) and a point, $w^* \in \tilde{\mathcal{W}}$, there exists a feedback law $g: \mathcal{W} \to \mathcal{U}$ such that w^* is a fixed point of the closed-loop map, f_g (7), if and only if*

(i) $w^* \in \mathcal{J}$;

(ii) $g(w^*) = u^* \triangleq \begin{bmatrix} -2/\gamma \\ -(1-\alpha)/(1+\alpha) \end{bmatrix} \dot{b}_2^*$

This result shows, on the one hand, that only a point in $\mathcal{J}$ may be fixed by feedback and, on the other hand, that an appropriate constant u^* may be found to fix *any* point of $\mathcal{J}$.

Local Stabilizability of the Task Plane

Next, observe that the system is locally controllable at any point in the vertical one-juggle task set.

Proposition 2 ([4]) *If*

$$w^* \in \mathcal{J}$$

and g fixes w^*, $f[w^*, g(w^*)] = w^*$, then system (6) is locally controllable at $[w^*, g(w^*)]$.

Local controllability, of course, implies local stabilizability. For, according to linear control theory, if (A, B) is a completely controllable pair, then for any desired set of poles whose complex elements appear in conjugate pairs, $\Lambda = \{\lambda_i\}_{i=1}^4 \subset \mathbb{C}$, there exists a matrix, $K_\Lambda \in \mathbb{R}^{2\times 4}$ such that the closed-loop spectrum achieves that set, *spectrum* $(A + BK_\Lambda) = \Lambda$.

Now suppose that A and B denote the fixed system and control input matrices resulting from a local linearization analysis of (6) around a desired set point, w^*—that is, $A = [D_w f](w^*, u^*)$; $B = [D_u f](w^*, u^*)$. If the robot feedback algorithm, g, is chosen to be

$$g(w) \triangleq u^* + K_\Lambda(w - w^*) \tag{8}$$

it follows that any K_Λ for which $\Lambda \subset \mathcal{D}^1 \subset \mathbb{C}$ (the open unit disk in the complex plane) yields a feedback law, g, which achieves the vertical one-juggle as defined earlier in this section. Thus, Proposition 2 demonstrates that the vertical one-juggle is logically achievable.

Robot Implementation

The preceding analysis employed a geometric representation of the task domain in terms of a discrete dynamical control system on puck velocities over the contact set, $\mathcal{I}$. That analysis permitted a rigorous definition of the task at hand and the logical assurance of its possibility. Attention now turns to the *robot control problem:* the synthesis of robot control laws guaranteed to result in impact schedules that accomplish a specified task. To this end, we will introduce a new geometric synthesis procedure defined on the entire cross-product puck-robot phase space, $\mathcal{W} \times \mathcal{B}$, which represents the continuous physical trajectories of both rather than the discrete-time evolution of their mutual impacts. This synthesis procedure gives rise to a family of robot control algorithms, which we demonstrate empirically and prove mathematically to be correct.

Our resort to this new controller geometry is a consequence of the unexpected failure of robot controllers based upon the straightforward algorithm (8). In an earlier paper [6], we gave a detailed explanation for this empirical result. Roughly speaking, this failure is attributed, on the one hand, to limitations of local linear controller design and, on the other hand, to the inappropriateness of the discrete task geometry for physically viable implementations in the real world of continuous trajectories and torque actuators.

This section describes the implementation of a successful vertical one-juggle on the physical apparatus and reviews the formal proof of its correctness summarized from the account in [4]. We first introduce the "mirror algorithm" and its relationship to the geometry of the continuous puck-robot phase space, $\mathcal{W} \times \mathcal{R}$. In order to prove its correctness, we then relate the continuous-time geometry of the mirror algorithm to the discrete-time geometry of the environmental control system. Finally, we present data from the physical experiments performed using this algorithm.

The Mirror Algorithm and Its Implementation

This section introduces a control procedure grounded in the robot's continuous-time framework. Two different ideas are at work. First, one "translates" the desired impact sequence into an analytic function of the continuously changing state of the environment. Thus, the robot is merely required to track a reference trajectory generated on-line from a mechanical system. This promises to be more robust than a procedure relying explicitly on state measurement at impact. The second idea, borrowing from Raibert [1], [7], is to use the total energy of the environment in this function in order to stabilize the motion around a desired trajectory. If the desired behavior is periodic, then it may be characterized at any instant—for example, just before impact—by specifying some constant value for desired total energy.

To better convey the intuitive origins of the new algorithm, we will first discuss the problem of a prismatic one-degree-of-freedom robot in a one-degree-of-freedom environment and then suggest how our solution scales to the present apparatus.

A Prismatic Robot in a One-Degree-of-Freedom Environment Suppose there is a single puck (of unit mass) constrained to fall in only the vertical direction and a piston that moves up and down to strike it precisely. The objective is to bring the puck to a specified periodic orbit via successive impacts. Suppose that r and b measure the height of the robot and the puck, respectively. Notice that the trivial "mirror" law

$$r = -\kappa_{10} b,$$

$$\kappa_{10} = -(1 - \alpha)/(1 + \alpha)$$

already satisfies the fixed point conditions of Proposition 1. Assuming the robot is tracking accurately, impact is guaranteed to occur at zero height. The vertical puck trajectory is completely determined by its total vertical energy,

$$\eta(b, \dot{b}) = \tfrac{1}{2}\dot{b}^2 + \gamma b$$

This fact can be conveniently used to enhance κ_{10} by a vertical energy error term to stabilize the above system at a fixed point:

$$\begin{aligned} r &= -\{\kappa_{10} + \kappa_{11}[\eta_d - \eta(b, \dot{b})]\} b \\ &= -\kappa(w) b \end{aligned} \tag{9}$$

The reader may note that an impact occurs either in the case of exact tracking $r(t) \equiv b(t)$ or when both the robot and the puck achieve zero height. In the latter case, the robot describes a distorted "mirror" reflection of the puck's trajectory. The reader may note as well that since friction is neglected during the puck's flight, we may assume that $\dot{\eta} \equiv 0$; hence,

$$\dot{r} = -\kappa(w)\dot{b}$$

Since the energy error term stays constant during flight (assuming, as we do, the absence of friction) in the tracking case, the robot would track the object during the entire flight. This can be considered a limiting case for κ_{11}.

Thus, this procedure makes explicit use of the full puck trajectory, completely specifies the robot's behavior, and results in a probable correct vertical one-juggle, when the robot and environment have the same "Cartesian" degrees of freedom.

A Revolute Robot in a Two-Degree-of-Freedom Environment This idea "scales" to the particular case at hand—the two-degree-of-freedom Cartesian environment presented earlier.

The basic idea carries over into this environment by just adding linear PD feedback compensation terms for the horizontal component. Define the "puck angle" as $\theta(b) = a \tan (b_2/b_1)$.

Now, as opposed to controlling the robot height as a function of puck height, we control the robot angle q as a function of puck angle:

$$q = -k_1(w)\theta + k_2(w)$$

In the particular set of experiments reported here, we have chosen

$$\begin{aligned} &\kappa_1(w, w^*) \\ &\quad \triangleq \kappa_{10} + \kappa_{11}[\eta(w^*) - \eta(w)] \\ &\kappa_2(w, w^*) \\ &\quad \triangleq \kappa_{21}(b_1 - b_1^*) + \kappa_{22}(\dot{b}_1 - \dot{b}_1^*) + \frac{1}{2} \\ &\qquad \cdot \frac{(w - w^*)^T M (w - w^*)}{[\kappa_{31} + ((w - w^*)^T M (w - w^*))^2]^2} \\ &\qquad + \frac{\kappa_{41}(\dot{b}_2 - \dot{b}_2^*)}{[\kappa_{42} + (\dot{b}_2 - \dot{b}_2^*)^2]^2} \end{aligned} \tag{10}$$

where k_{ij} are fixed constant gains, and M is a symmetric matrix in $\mathbb{R}^{4\times 4}$.

The first two terms in k_2 are borrowed from standard linear feedback control theory, implementing proportional derivative feedback. Analyzing the linearized system at a fixed point with just these two terms in k_2 results in an ill-conditioned system—the controllability matrix is nearly singular. The last two terms in this expression were introduced to ensure complete controllability locally without confounding the favorable global properties of the algorithm.

Analytical Results

This summary of our analytical results is taken from the complete presentation in [4], to which the reader should refer for a more complete discussion as well as all proofs.

When the robot has achieved the reference "mirror" trajectory described above, then the puck and robot trajectories lie on a "mirror surface," $\mathfrak{M} \subseteq \mathcal{W} \times \mathcal{R}$, in the cross product phase space,

$$\begin{aligned} \mathfrak{M} \triangleq \{&[(b, \dot{b}), (q, \dot{q})] \in \mathcal{W} \times \mathcal{R}: (q, \dot{q}) \\ &= [\mu(b, \dot{b}), (D\mu \cdot n)(b, \dot{b})]\} \end{aligned}$$

specified as the graph of the function $\mu(w) \triangleq \kappa_1(w) \cdot \theta - \kappa_2(w)$, and its derivative along the motion of the puck (2). Recall θ is the "puck angle" and κ_i are the gain functions detailed in (10).

Examination of the intersection between $\mathfrak{M}$ and the velocities over the contact set, $\mathcal{I}$, reveals how to choose the gains in κ_1 (10) to achieve the fixed point conditions of Proposition 1 for any $w^* \in \mathfrak{I}$. A central result [4, Proposition 5.2] shows via projection of this intersection onto $\mathcal{W}$ that the robot's "mirroring" motion induces a three-dimensional invariant submanifold of the environmental control system (6). In consequence [4, Corollary 5.3], the local stability behavior of any valid vertical one-juggle task, w^*

$\in \mathfrak{I}$, may be adjusted by the appropriate choice of gains in κ_2 (10).

Empirical Results

We now present plots of simulated and experimental data in order to validate our simplified model used for analysis and to illustrate the utility of our analytical results for both the idealized model as well as the real system.

Figure 3, a "recording" of a successful vertical one-juggle using all parameters derived from analytical procedures, nicely depicts the rapid convergence for initial conditions (in drop-off position) from any region within the puck's workspace not too close to the origin—a kinematic singularity. Despite departures from the idealized model and the relatively large sensor noise discussed in the beginning of this section, it may be observed from this and the subsequent plots that our algorithm produces steady reliable juggling performance. We have recorded vertical one-juggle runs with hundreds of impacts without encountering any failures.

Figures 4 and 5 compare the responses of the analytical model with and without friction to the responses of our experimental setup for two different initial conditions. Each experimental data curve displays statistical information (mean plus/minus one standard deviation) obtained from 20 successive runs (without handpicking). This presentation promises to offer a closer rendering of true performance than one based upon a handpicked best run. The steady-state values in the horizontal impact position, b_1, are very close around the desired value for both curves. The plots of the vertical impact velocity, $\dot{b}_2$, demonstrate, first, as we expected, that the effect of the unmodeled friction is a steady-state deviation, which, second, is rather accurately predicted by the one-degree-of-freedom model that includes friction. In examining the transients, notice that the experimental transient responses for $\dot{b}_2$ (lower plots) consistently match the responses of the model with friction, as expected. However, for b_1 (upper plots), the experimental transient responses are closer to the much faster transient model responses *without* friction than to those of the model with friction. This favorable discrepancy is not completely understood at present. We suspect that a lower friction value at the low horizontal velocities, or the unmodeled effect of spin on the impact, for example, might be responsible for this benign discrepancy.

Recall that the last two terms in (10) were introduced to arbitrarily specify the local behavior—that is, to place the poles of the linearized system. However, no significant changes were observed experimentally: the effects of the linear terms are apparently very small and disappear in the measurement noise, or are dominated by unmodeled dynamics. This corroborates our comments in [6] concerning the relative insignificance of local stability properties in the present setting.

Conclusion

We have shown how the geometry of the impact configurations, $\mathfrak{I}$, leads to a discrete dynamical model of the effect of robot impact strategies upon the behavior of an otherwise free-falling puck. This model provides a framework for rigorously defining dexterous robotic tasks—for example, the "vertical one-juggle"—and determining their feasibility. Although prescriptions for explicit impact strategies may be extracted from this model as well, it does not seem to offer an empirically viable framework for synthesis of robot control laws.

After many failed attempts to implement a logically correct but physically underconstrained and nonrobust algorithm extracted from the discrete dynamics arising out of this "contact geometry," we were led to a new type of control algorithm based on a completely different "mirror geometry" inhabiting the continuous phase space of the robot-environment pair. Experiments attest to the effectiveness of this control design. Moreover, analysis of the intersection between the mirror surface and the impact surface results in a correctness proof with respect to the discrete dynamical "environmental control system" that formally defines the task.

The central notion of robot controller synthesis via a "mirror geometry" in phase space appears to generalize to other interesting robotic tasks in this domain. For example, we have extended it to the task of catching falling objects and have applied it successfully to the task of juggling two pucks simultaneously as well. Retrospective correctness proofs notwithstanding, the generation of algorithm geometry is completely heuristic at present: each synthesis is empirically hand-tailored to fit the given task. Nevertheless, the analytical tractability of the resulting robot-environment closed loop as demonstrated here raises the hope that sufficient understanding may soon be realized to afford automatic translation of suitably expressed task definitions into provably correct and empirically valid robot controller designs.

In the longer term, we believe these ideas will have still wider application. For example, analytical techniques similar to those employed here result in correctness proofs for (simplified versions of) Raibert's empirically verified legged locomotion algorithms [7]. Our juggler and Raibert's hopper "settle down" to a characteristic steady-state pattern because that pattern is an attracting periodic orbit of the closed-loop robot-environment dynamics. Very likely, similar "natural" control mechanisms would make good candidates for gait regulation and other more complex tasks requiring controlled intermittent collisions with a dynamical environment.

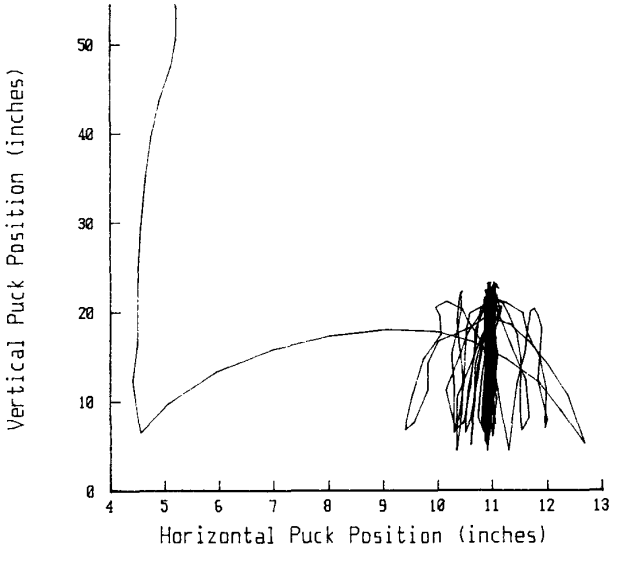


Fig. 3. Sample continuous data.

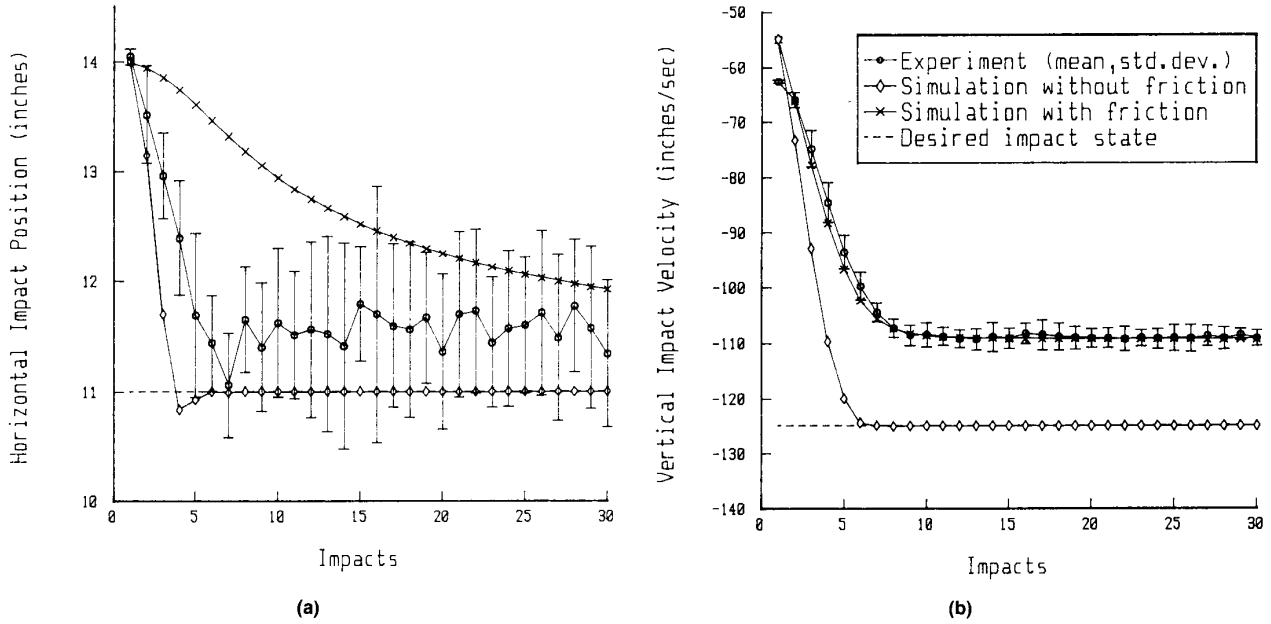


Fig. 4. Initial puck position at the lower right on the juggling plane.

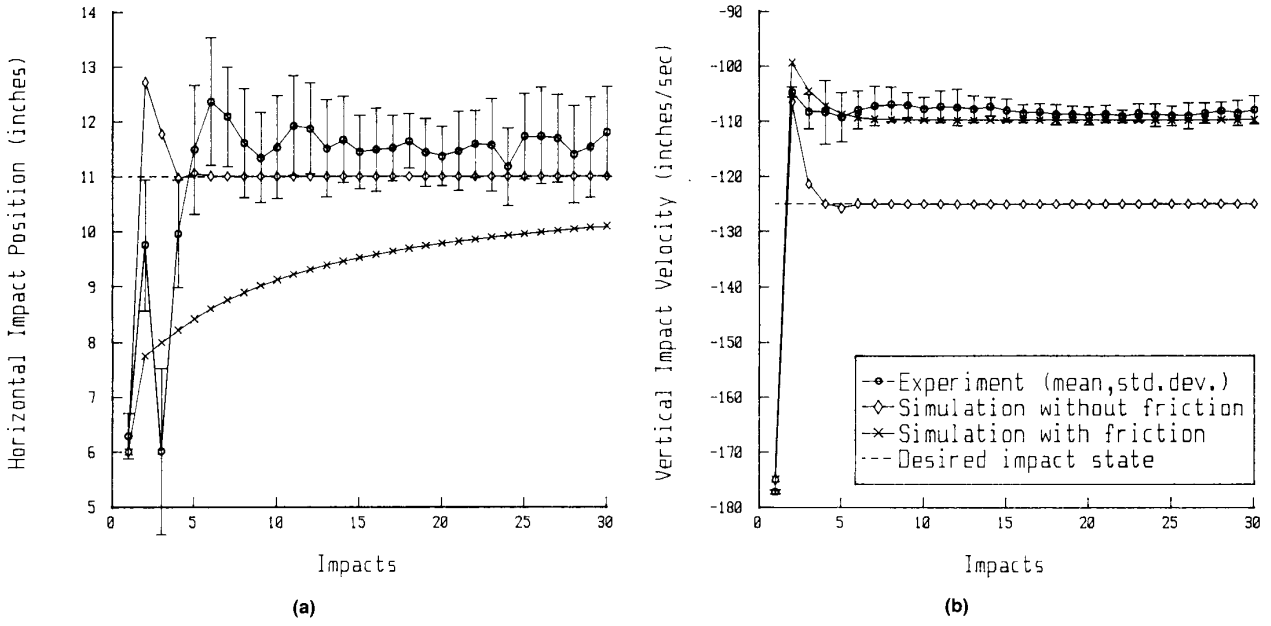


Fig. 5. Initial puck position at the upper left on the juggling plane.

Acknowledgments

This work has been supported in part by PMI Motion Technologies, GMF Robotics Corporation, INMOS Corporation, and the National Science Foundation under a Presidential Young Investigator Award held by the second author.

References

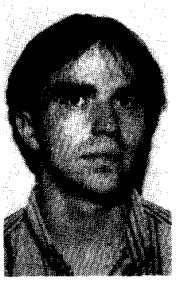
[1] Marc H. Raibert, *Legged Robots That Balance*, Cambridge, MA: MIT Press, 1986.

[2] F. Levin, M. Bühler, and D. E. Koditschek, "The Yale Real-Time Distributed Control Node," *Second Annual Workshop on Parallel Computing*, Oregon State University, Portland, OR, Apr. 1988.

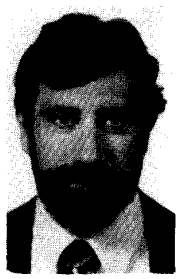
[3] M. Bühler, C. J. Taylor, N. Vlamis, and A. Ganz, "The Cyclops Vision System," *Proc. North American Transputer Users Group*, pp. 141-149, Salt Lake City, UT, Apr. 1989.

[4] M. Bühler, D. E. Koditschek, and P. J. Kindlmann, *Robotics in Intermittent Dynamical Environments*, Technical Report 8812, Center for Systems Science, Yale University,

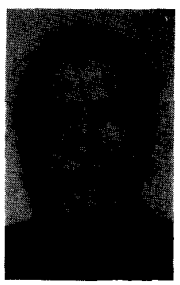
June 1988 (Revised Mar. 1989).
[5] J. L. Synge and B. A. Griffith, *Principles of Mechanics*, London: McGraw-Hill, 1959.
[6] M. Bühler, D. E. Koditschek, and P. J. Kindlmann, "A One Degree of Freedom Juggler in a Two Degree of Freedom Environment," *Proc. IEEE Conf. Intelligent Syst. and Robots*, pp. 91–97, Tokyo, Japan, Oct. 1988.
[7] M. Bühler and D. E. Koditschek, "Analysis of a Simplified Hopping Robot," *IEEE Intl. Conf. Robotics and Automat.*, pp. 817–819, Philadelphia, PA, Apr. 1988.



Martin Bühler studied electrical engineering at the University of Karlsruhe, West Germany, from 1981 until 1983. In 1985, he received his M.Sc. degree from the Electrical Engineering Department at Yale University, where he is presently completing his Ph.D. thesis, entitled "Robotics in Intermittent Dynamical Environments."



Dan E. Koditschek received his B.S. degree in 1977 and his Ph.D. degree in 1983 from the Department of Electrical Engineering at Yale University. He has been a member of the faculty of the Yale Electrical Engineering Department ever since and presently holds the rank of Associate Professor. His present interests largely concern the applications of dynamical systems theory to robotics.



Peter J. Kindlmann is an Associate Professor (Adjunct) of Electrical Engineering at Yale University. He received the B.S. degree in physics from Columbia College and the M.S. and Ph.D. degrees in electrical engineering from Yale University. At Yale, he was the Founder and Director of Yale's Instrumentation Lab and Director of Industrial Liaison for Engineering and Applied Science. Now he divides his time between Yale and Congruent Design, Inc., a consulting R&D company in the field of instrumentation and product design of which he is Founder and President. Current interests include teaching and the development of new methods of measurement and automation in "difficult" areas, such as high-voltage systems, metal casting, and biotechnology.

1990 ACC

The American Automatic Control Council will hold the ninth American Control Conference (ACC) Wednesday through Friday, May 23–25, 1990, at the Sheraton Harbor Island Hotel, San Diego, California. This conference will bring together people working in the fields of control, automation, and related areas.

Both contributed and invited papers will be included in the program. The ACC will cover a range of topics relevant to the theory and practical implementation of control and industrial automation and to university education in controls. Topics of interest include, but are not limited to, the following: linear and nonlinear systems, identification and estimation, signal processing, multivariable systems, large-scale systems, robotics and manufacturing systems, guidance and control, sensors, simulation, adaptive control, optimal control, expert systems, and control applications.

The Organizing Committee plans to arrange workshops to be held in conjunction with the 1990 ACC.

For further information, contact the General Chairman:

Dagfinn Gangsaas
Bocing Advanced Systems
P.O. Box 3707, M/S 33-12
Seattle, WA 98124-2207
Phone: (206) 393-8779

Adjacent to San Diego is La Jolla's rugged, beautiful coastline blending into lushly covered hillsides, which has often been compared with the French Riviera. Further inducements of this seaside village include quaint shops and restaurants, balmy breezes, a variety of water sports from surfing to diving, and championship golf and tennis facilities.

Out of Control

"A non-smoking window, in the right half of the plane, please!"