

Department of Electrical & Systems Engineering

Departmental Papers (ESE)

University of Pennsylvania

Year 1990

Globally stable closed loops imply
autonomous behavior

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

Copyright 1990 IEEE. Reprinted from *Proceedings of the 5th IEEE International Symposium on Intelligent Control*, Volume 2, 1990, pages 651-656.

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, author Daniel Koditschek was affiliated with Yale University. Currently, he is a faculty member in the Department of Electrical and Systems Engineering at the University of Pennsylvania.

This paper is posted at ScholarlyCommons.
http://repository.upenn.edu/ese_papers/387

Globally Stable Closed Loops Imply Autonomous Behavior

Daniel E. Koditschek *

Center for Systems Science
Yale University, Department of Electrical Engineering

Abstract

An autonomous machine can operate successfully in a diversity of situations without resort to intervention by “higher level” processes, for example, humans. Physical machines are ultimately force or torque controlled dynamical systems: the specification of input torques, whether via syntactic prescriptions or feedback controllers, results in certain classes of vector fields. Control procedures whose resulting vector fields have globally attracting goal states may properly be said to evince autonomous behavior.

This paper reviews various procedures developed within the Yale Robotics Lab that result in provably autonomous behavior according to the criterion developed above. Simulation results and physical experimental studies suggest the practicability of these methods.

1 Introduction

This paper reviews a program of research in robotics that seeks to encode abstract tasks in a form that simultaneously affords a control scheme for these torque actuated dynamical systems as well as a proof that the resulting closed loop behavior will correctly achieve the desired goals. Two different behaviors that require dexterity and might plausibly connote “intelligence” — navigating in a cluttered environment, and juggling a number of otherwise freely falling objects — are examined with regard to similarities in problem representation, method of solution, and causes of success. The central theme of the paper concerns the virtue of global stability mechanisms. They lend autonomy — that is, freedom from dependence upon some “higher” intelligence — at the planning level. They encourage the design of “canonical” procedures for “model” problems which may then be instantiated in particular settings by a change of coordinates.

1.1 Representation

The Robot Key to this point of view is the fact that any machine operating in the physical world is subject to both dynamical as well as geometric constraints. Kinematic chains impose a conceptually straightforward [18] but mathematically complicated [34] geometry. Their Newtonian dynamics result in strongly nonlinear and surprisingly complex equations [4]. Yet since

these constraints have been understood and satisfactorily modeled for a long time, well posed control problems for robot “plants” may be stated relatively easily. For purposes of the present paper, take the standard Lagrangian model of the “plant” that arises from rigid body assumptions.

The Environment In all but the most trivial instances, the robot’s desired behavior involves interaction with an environment, \mathcal{E} , that must itself possess geometric and dynamical properties. Moreover, in the context of particular tasks, various aspects of the robot’s operation in the environment will give rise to a new set, \mathcal{P} , that might be called the “planning set”, within which particular goals may be formally represented. For example, in the case of robot motion planning, the environment’s geometry together with the robot’s kinematics and geometry give rise to a freespace [23] within which the robot’s motions are constrained to lie. For peg-in-hole [39] and related problems of pushing [25, 24] additional structure must be added to the geometry of this space [6, 15] to account for frictional and jamming (contact degrees of freedom) reaction forces that the environment may impose upon the robot. Tasks such as playing ping-pong [2] or walking and running [29, 26] require explicit attention to the dynamics as well as the geometry of the environment, and thus the planning set will consist of a space and a dynamical system operating on it as will be illustrated below.

The Task Finally, a robot operating in a specified environment might be assigned a variety of tasks. The specific task desired — an abstraction meaningful initially only to its human originator — must be encoded in terms that relate to the robot in its environment. Thus, within the context of the planning set there must be devised a formal representation of the desired behavior — the “encoding”. In the two example task domains examined here the encoding takes the form of a goal state, $\mathcal{G} \in \mathcal{P}$ to which it is desired to bring the robot-environment pair. Ultimately, in order to achieve the task, some controller must be specified for the plant: it must ensure simultaneously that the robot both achieve the task as well as respect the environmental constraints. As the robot’s (and, possibly, the environment’s) state changes under the action of this controller, the planning space reflects these changes. Apparently, then, the solution to a robotic task imposes a dynamical system upon \mathcal{P} . When, as here, the task at hand admits the representa-

*This work was supported in part by the National Science Foundation under grant DMC-8505160, and, in part by a Presidential Young Investigator Award.

tion in the form of a goal state(s), \mathcal{G} , then a successful control scheme is one whose associated dynamics on \mathcal{P} brings as many initial states to \mathcal{G} as possible.

1.2 Intelligent Robots and Intelligent Designers

Intelligent Behavior Connotes Autonomy

This paper holds autonomy to be a primary design objective in the construction of intelligent machines. A machine equipped with an intelligent strategy ought to be able to contend with the full spectrum of logically possible circumstances that arise in completion of its task.

Translated into the context of dynamical systems theory, autonomy implies global convergence. That is, an intelligent control strategy ought to be capable of bringing the robot-environment pair to the goal states, \mathcal{G} , from any initial state represented in the planning space. More succinctly, \mathcal{G} ought to be an attracting set whose domain of attraction is \mathcal{P} . Unfortunately, this is not always possible: Section 2 describes a situation where such global properties are topologically impossible. Instead, one might imagine a situation where “almost all” initial states are brought to the desired goal, and what is left over is very small. If the domain of attraction of a locally attracting set, \mathcal{G} , includes all but a set of measure zero then say that \mathcal{G} is *essentially globally asymptotically stable*. Even where no topological obstructions are present, and even if one settles for essential global properties it is an unfortunate fact that estimating the domain of attraction of locally attracting sets is very difficult in practice.

For linear dynamical systems on a vector space, a local computation involving the eigenvalues of a matrix affords global conclusions. This is the archetypally “practicable” means of assuring global properties, and is almost by definition not to be found in general in the nonlinear case. The two examples presented in this paper, however represent instances where a global stability mechanism does enjoy such a practicable property: namely, a series of locally defined computations involving jacobians and their eigenvalues.

Intelligent Design Connotes Generalization

If global stability properties are a primary objective in the design of intelligent robot controllers, yet practicable instances of such mechanisms are rare, then an intelligent designer will seek to use and re-use existing instances again and again. A second theme of this paper is the utility of generalizing a specific controller design through a change of coordinates. The two behaviors reported below are achieved by recourse to two different stability mechanisms which share the unusual property of practicability in the sense developed above. In each case, the paper attempts to show how a canonical solution to a simple problem can be “deformed” into the solution to a seemingly complicated but essentially equivalent problem via the appropriate transformation of the problem space.

1.3 A Program of Robotics Research

It is not at all clear how to tell a robot to “fold the laundry” or “scramble the eggs” or “make the bed”. For such tasks neither the environment, \mathcal{E} , nor the appropriate planning space, \mathcal{P} , nor the task encoding, \mathcal{G} , seem very obvious. The program of research reviewed in this paper seeks to make progress toward the analysis and achievement of such confusing robotic tasks by

a methodical investigation of more straightforward examples. It should be immediately emphasized that the only thing straightforward about these examples is the conceptual distinction between task, environment, and robot. Navigation has been shown to be fundamentally difficult [35], and juggling has not even been attempted until recently [9, 1]. Yet in these cases it has seemed sufficiently clear how to disentangle the constituent pieces of the problem definition that a careful look at how they interact in a successful implementation might provide more general insight into the problem of task encoding.

2 An Essentially Global Navigation Scheme

Let a robot move in a cluttered but perfectly known workplace. There is a particular location of interest and it is desired that the robot move to this location from anywhere else in the workplace without colliding with the obstacles present.

2.1 Representation

The constituent pieces of the problem seem readily apparent in this case. The robot model has already been introduced. The environment, \mathcal{E} , is simply the workplace — a subset of Euclidean 3-space remaining after the obstacles are removed. Contained within the robot’s configuration space is the *free space*, \mathcal{F} — the set of all robot placements which do not involve intersection with any of the “obstacles” cluttering the workplace. The appropriate planning set, \mathcal{P} , for this problem is now clear: it is the phase space formed over \mathcal{F} , that is, the union of all the robot’s configuration space velocity vectors taken over each configuration in \mathcal{F} . For present purposes this may be modelled as a smooth manifold with boundary (but see [32] for the case of sharp corners). The task also seems straightforward to represent: a particular navigation problem results from the choice of one particular destination point in the interior of the freespace. The goal set, \mathcal{G} , is a singleton: the destination point at zero velocity. The problem is now to find a feedback controller under whose influence the robot’s state will approach \mathcal{G} from as large a set of initial configurations as possible while remaining in \mathcal{P} .

2.2 Navigation Functions

Initiated by Khatib a decade ago [19], the idea of using artificial potential functions for robot task description and control was adopted or re-introduced independently by a number of researchers [27, 3, 28]. Since the interest in artificial potential functions originally emerged within the robotic control community, it is perhaps not surprising that little attention was paid to the algorithmic issues of global path planning in this literature. The question of whether the method could be used to guarantee the construction of a path between any two points in a path-connected space remained unexplored. Yet it is exactly this kind of global property that would lend autonomy from “higher level” intelligence to the controller.

A Practicable Global Stability Mechanism

In the present context, the utility of artificial potential functions for path planning rests upon the possibility of deducing global stability properties from local computations. Because the potential function serves as a global Lyapunov function for its gradient vector field, it is easy to see that the minima of a gradient system

(that satisfies certain regularity conditions) will attract almost all trajectories [17, 21]. Of course, the condition for a minimum is a local one that may be constructively checked via calculus and algebraic computation. Thus, if it can be assured that there is only one minimum and that it coincides with the desired destination then a potential function serves as a global path planner on the freespace, \mathcal{F} . Of course, the appropriate planning space is \mathcal{P} , the space of legal configurations and all their possible velocities. But a slight extension to Lord Kelvin's century old results on energy dissipation suffices to make the same machinery work with a suitably designed controller for the plant on \mathcal{P} [21].

Existence Gradually, there seems to have emerged a common awareness of several fundamental problems with the potential function methodology. Spurious local minima seemed unavoidable, and unrealizable infinite torques were thought to be required at the obstacle boundaries. In fact, an artificial potential function need satisfy a longer list of technical conditions in order to give rise to a bounded torque feedback controller that guarantees convergence to the goal state, \mathcal{G} , from almost every initial configuration. This list comprises the notion of a *navigation function* introduced to the literature two years ago [30].

The question immediately arises whether such desirable features may be achieved in general. In fact, the answer is affirmative: smooth navigation functions exist on any compact connected smooth manifold with boundary [22]. Thus, in any problem involving motion of a mechanical system through a cluttered space (with perfect information and no requirement of physical contact) if the problem may be solved at all, we are guaranteed that it may be solved by a navigation function. There remains the engineering problem of how to construct such functions

Invariance The importance of coordinate changes and their invariants is by now a well known theme in control theory. Roughly speaking, these notions formalize the manner in which two apparently different problems are actually the same. Their most familiar instance is undoubtedly encountered in the category of linear maps on linear vector spaces whose invariants (under changes of basis) determine closed loop stability. Of course, many other instances may be found in the control literature and, more recently, the utility of coordinate changes in robotics applications has been proposed independently by Brockett [5] as well.

The relevant invariant in navigation problems is the topology of the underlying configuration space [20]. In this regard, the significant virtue of the navigation function is that its desirable properties are invariant under diffeomorphism [22]. Thus, instead of building a navigation function for each particular problem, we are encouraged to devise "model problems", construct the appropriate model navigation functions, and then "deform" them into the particular details of a specified problem.

2.3 The Construction of Navigation Functions

A "Model" Problem A "Euclidean sphere world" is a compact connected subset of \mathbf{E}^n whose boundary is the disjoint union of a finite number, say $M + 1$, of $(n - 1)$ -spheres. We suppose that perfect information about this space has been furnished in the form of $M + 1$ center points $\{q_i\}_{i=0}^M$ and radii $\{\rho_i\}_{i=0}^M$ for each of

the bounding spheres. Let the "bad" set of obstacle boundaries to be avoided be encoded by the product function, $\beta : \mathcal{M} \rightarrow [0, \infty)$ is,

$$\beta \triangleq \prod_{i=0}^M \beta_i,$$

where β_i vanishes on the i^{th} sphere.

Theorem 1 ([22]) *Let the freespace, \mathcal{F} be a Euclidean sphere world. Then there exists a positive integer N such that for every $k \geq N$, φ ,*

$$\varphi = \sigma_d \circ \sigma \circ \hat{\varphi} = \left(\frac{\gamma^k}{\gamma^k + \beta} \right)^{\frac{1}{k}}, \quad (1)$$

is a navigation function for destination $q_d \in \mathcal{F}$, where

$$\gamma \triangleq \|q - q_d\|^2.$$

The function, N , on which the theorem depends is given explicitly in [22].

A Class of Coordinates Transformations A *star shaped set* is a diffeomorph of a Euclidean n -disk, \mathcal{D}^n possessed of a distinguished interior *center point* from which all rays intersect its boundary in a unique point. A *star world* is a compact connected subset of \mathbf{E}^n whose boundary is the disjoint union of a finite number of star shaped set boundaries. Now suppose the availability of an implicit representation for each boundary component: that is, let β_i be a smooth scalar valued function that vanishes on the boundary of the i^{th} obstacle, as before. Assume, moreover, that a known center point location, q_j has been specified for each obstacle as well. Further geometric information required in the construction to follow is detailed in the chief reference for this work [31]. A suitable Euclidean sphere world model, \mathcal{M} , is explicitly constructed from this data. That is, one determines (p_j, ρ_j) , the center and radius of a model j^{th} sphere, according to the center and minimum "radius" (the minimal distance from q_j to the j^{th} obstacle) of the j^{th} star shaped obstacle. The *star world transformation* is now given as

$$h_\lambda(q) \triangleq \sum_{j=0}^M \sigma_j(q, \lambda) [\nu_j(q) \cdot (q - q_j) + p_j] + \sigma_d(q, \lambda) [(q - q_d) + p_d], \quad (2)$$

where σ_j is the j^{th} *analytic switch*, and ν_j is the j^{th} *star set deforming factor* (see [31] for the explicit formulae). The "switches", make h look like the j^{th} deforming factor in the vicinity of the j^{th} obstacle, and like the identity map away from all the obstacle boundaries. With some further geometric computation we are able to prove the following.

Theorem 2 ([31]) *For any valid star world, \mathcal{F} , there exists a suitable model sphere world \mathcal{M} , and a positive constant Λ , such that if $\lambda \geq \Lambda$, then*

$$h_\lambda : \mathcal{F} \rightarrow \mathcal{M},$$

is an analytic diffeomorphism.

Thus, if φ is a navigation function on \mathcal{M} , the construction of h_λ automatically induces a navigation function on \mathcal{F} via composition, $\tilde{\varphi} \triangleq \varphi \circ h_\lambda$.

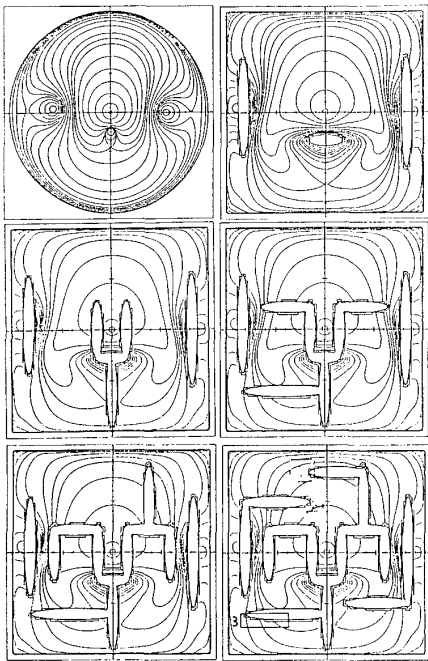


Figure 1: Planar forest of stars with three internal tree-like obstacles (bottom right), its “purged” versions, and its model sphere world (top left). [32]

Navigation Functions for Geometrically Complicated Spaces In a recent paper [32], we show how to extend significantly this class of coordinate transformations. Consider the scene depicted in the lower right hand corner of Figure 1 resembling a building floor plan (the results, of course, work in arbitrary dimensions) that we call a “forest of stars.” There are three internal tree-like obstacles, and the depth of the deepest tree is $d = 4$. According to the method described in [32] a “purging transformation,” $f_{\Lambda, d}$, is applied d times, until a space whose obstacles are the roots of the original trees is obtained. This space is a star world: the the previously constructed star-world to sphere-world transformation [31] is now be used to obtain the corresponding model sphere world, \mathcal{M} , on which the simple navigation function may be used.

3 An Essentially Global Juggler

This section presents work in a task domain requiring dynamical dexterity. While our analytical understanding here is much less mature than in the previous section, the unity of methodological pursuit will hopefully become apparent.

3.1 The Yale Juggler

Robot and Environment Models The physical apparatus consists of a puck, which slides on an inclined plane and is batted successively by a simple “robot:” a bar with billiard cushion rotating in a juggling plane.

In isolation, the robot’s dynamics occur in its phase space, $\mathcal{V} \triangleq \mathcal{R} \times \mathbb{R}$, of angular positions and velocities, and may be modeled by the standard double integrator forced by commanded torque. In isolation, the puck’s

dynamics occur in its phase space, $\mathcal{W} \triangleq \mathcal{B} \times \mathbb{R}^2$, and may be modeled simply by the equations of free flight in the earth’s constant gravitational field. Thus note in contrast to the previous situation that \mathcal{E} , the environment, is represented by the pair of phase space, \mathcal{W} , and the ballistic dynamical model operating upon it.

The Planning Space The coupling of robot and puck dynamics may be represented by a collision map that takes the à priori puck-robot states at contact, into the new puck velocity vector after contact [9, 11, 8]. The future trajectory of the body in \mathcal{W} subsequent to an impact event is readily derived by integrating the free flight model starting with the initial conditions just after impact. The robot can determine where (or, equivalently, after how much elapsed time since the previous impact, t , now denoted u_1) to hit the puck and with what linear velocity ($\|b\|\dot{r}$, now denoted u_2) the impact should occur. In the mean time, the puck’s behavior cannot be altered. On the most fundamental level, the robot’s two actions, $u = [u_1, u_2]^T \in \mathcal{U}$, represent the only means of imposing control upon its environment. Thus, we treat the robot as an independent external “agent of control” and consider the various puck behaviors resulting from the robot’s actions,

$$w_{j+1} = f(w_j, u_j).$$

The planning set, \mathcal{P} , then is the dynamical system $f: \mathcal{W} \times \mathcal{U} \rightarrow \mathcal{W}$.

The Task 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, we want to be able to specify an arbitrary “apex” point together with a vertical position, and from arbitrary initial puck conditions, force the puck to attain a periodic trajectory which impacts at zero vertical position and passes through that apex point. We call this task the *vertical one-juggle*. Such tasks are exactly represented by the choice of a desired fixed point, w^* , now serves to define the goal state, \mathcal{G} .

Being interested in sensor based manipulation we focus upon solving such problems with feedback based controllers. Thus, a robot feedback strategy is a map¹, $g: \mathcal{W} \rightarrow \mathcal{U}$, from the body’s state to the robot’s action set, \mathcal{U} , resulting in the impact strategy $u_j = g(w_j)$.

3.2 The Stability Properties of Unimodal Maps

The juggling robot described above successfully juggles pucks falling (otherwise) freely on the (“frictionless”) juggling plane inclined into the earth’s gravitational field [12]. The robot’s “plan of action”, g , is accomplished by recourse to a family of “mirror laws” that map puck states into desired robot states at every instant of time. Extensions of the mirror idea have been

¹Although we prefer to avoid time varying controllers, there is no à priori objection to dynamical controllers. In practice, the memoryless control structure presented here suffices for all the tasks we have encountered to date.

shown experimentally to accomplish tasks such as juggling one and two pucks, and catching in a stable and robust manner [7]. Analytical results obtained to date suggest that these mirror algorithms owe their success to a new global stability mechanism quite different from the one explored in the previous section except in that it satisfies the critical criterion of “practicability” established in the introduction.

A Practicable Global Stability Mechanism

In contrast to the notion of energy dissipation that has been known for more than a century [38], the juggling behavior seems to arise through a stability mechanism that has been only recently recognized. The principal results required here were stated a little more than a decade ago by Singer [37] and Guckenheimer [16]. They studied the dynamical systems arising from iterations of a special class of maps on the unit interval into itself: the *S-unimodal maps*.

Singer showed that S-unimodal maps can have at most one attracting periodic orbit [37]. Guckenheimer showed that the domain of attraction of such attracting orbits includes the entire unit interval with the possible exception of a zero measure set [16]. Thus, an asymptotically stable orbit of an S-unimodal map is essentially globally asymptotically stable. In other words, a local computation at a candidate fixed point suffices to demonstrate its global stability properties.

Invariance Although the Singer-Guckenheimer theory is stated in terms of the apparently restrictive class of unit interval preserving maps, it extends to (at least) all their differentiable conjugates. Namely, say that g is a *smooth S-unimodal map* if there is an S-unimodal map, f , to which g is differentially conjugate — i.e. there exists a smooth and smoothly invertible function, h such that $g = h \circ f \circ h^{-1}$. It is straightforward to show that an attracting orbit of a smooth S-unimodal map is essentially globally asymptotically stable [14, 13].

Smooth S-unimodal maps form a sufficiently large family that this theory appears to have broad engineering applicability. For example, as described below, the line-juggler map falls within this class. Moreover, we have shown that simplified models of Raibert’s hopping robots give rise to smooth S-unimodal maps as well [10]. An important caveat is that the Singer-Guckenheimer theory at present has only limited extensions to higher dimensional systems. Thus, in all cases where we would like to invoke these results, we have had to restrict attention to simplified one degree of freedom models of the systems in question.

3.3 The Mirror Algorithm

For the purposes of this paper, it seems most convenient to limit discussion of the mirror algorithm to a simplified one-degree-of-freedom environment: “juggling” bead on a vertical wire. In any event, this is the model to which the Singer-Guckenheimer results are most directly applicable.

Construction Let the robot track exactly a continuous “distorted mirror” trajectory of the puck where, borrowing an idea from Raibert [29], the distortion factor is a function of the error in vertical total energy, η :

$$r = -\kappa_1(w)b; \quad \kappa_1(w) \triangleq \kappa_{10} + \kappa_{11}[\eta(w^*) - \eta(w)]. \quad (3)$$

Implementing a mirror algorithm is an exercise in robot trajectory tracking wherein the reference trajectory is a function of the puck’s state.

Analysis It is shown in [10] that the feedback law, g , resulting from the strategy described above may be determined in closed form (for the simplified one degree of freedom model). Substituting into the impact function yields the scalar map of puck impact velocities just before impact at the invariant impact position $b^* = 0$,

$$f(\dot{b}) = \dot{b} (1 - \beta(\dot{b}^2 - \dot{b}^{*2})), \quad (4)$$

where $\beta = \kappa_{11} \cdot (1 + \alpha)/2$.

It is not hard to show [10] that (4) satisfies the conditions of “S-unimodality” described above. A check of (4) reveals that the fixed point is locally stable when

$$0 < \beta < \frac{2/\zeta - 1}{\dot{b}^{*2}}. \quad (5)$$

There immediately follows,

Theorem 3 ([13]) *The mirror algorithm for the line-juggler results in a successful vertical one juggle which is essentially globally asymptotically stable as long as β satisfies the inequality (5).*

Experiments We have shown a gratifying correspondence between theoretical predictions based upon the Singer-Guckenheimer results, simulation studies, and physical data [10]. Perhaps the most dramatic depiction of this correspondence is suggested by our bifurcation studies, for which there is no space in the present paper. This section nevertheless provides some feeling for the predictive power of the theory described above. The experimental data in Figure 2 confirm that the transients can be predicted by recourse to local linear analysis of the scalar impact map. This behavior is confirmed even from large initial conditions (“globally”) on the juggling apparatus.

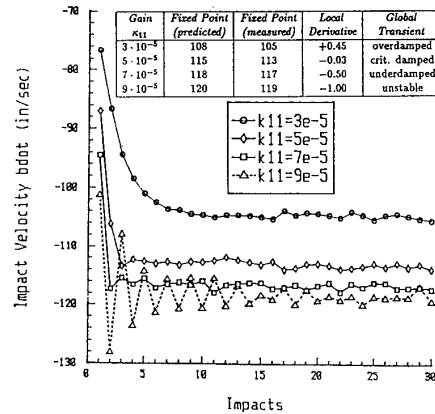


Figure 2: Line-Juggler Transients: Experimental Data

References

- [1] E. W. Aboaf, S. M. Drucker, and C. G. Atkeson. Task-level robot learning: Juggling a tennis ball more accurately. In *Proceedings IEEE International Conference on Robotics and Automation*, pages 1290–1295, Scottsdale, AZ, May 1989.
- [2] R. L. Andersson. Aggressive trajectory generator for a robot ping-pong player. In *Proceedings 1988 International Conference on Robotics and Automation*, pages 188–193, Philadelphia, PA, 1988.
- [3] J. R. Andrews and N. Hogan. Impedance control as a framework for implementing obstacle avoidance in a manipulator. In David E. Hardt and Wayne J. Book, editors, *Control of Manufacturing and Robotic Systems*, pages 243–251. A.S.M.E., Boston, MA, 1983.
- [4] Antal K. Bejczy. Robot arm dynamics and control. Technical Report 33-699, Jet Propulsion Laboratory, Pasadena, CA, 1974.
- [5] R. W. Brockett. On the computer control of movement. In *Proc. IEEE Conference on Robotics and Automation*, pages 534–540. IEEE, Philadelphia, PA., 1988.
- [6] Randy C. Brost. Computing metric and topological properties of configuration space obstacles. In *Proc. IEEE Conference on Robotics and Automation*, pages 170–176. IEEE, Scottsdale, AZ., 1989.
- [7] M. Bühler, D. E. Koditschek, and P.J. Kindlmann. Planning and Control of Robotic Juggling Tasks. In H. Miura, editor, *Fifth International Symposium on Robotics Research*, page (to appear). MIT Press, 1989.
- [8] M. Bühler, D. E. Koditschek, and P.J. Kindlmann. A Simple Juggling Robot: Theory and Experimentation. In V. Hayward and O. Khatib, editors, *Experimental Robotics I*, pages 35–73. Springer-Verlag, 1990.
- [9] M. Bühler and D. E. Koditschek. A prelude to juggling. In *26th IEEE Conference on Decision and Control*, pages (paper available from authors — not in proceedings), Los Angeles, CA., Dec. 1987.
- [10] M. Bühler and D. E. Koditschek. From stable to chaotic juggling. In *Proc. IEEE International Conference on Robotics and Automation*, page (to appear), Cincinnati, OH, May 1990.
- [11] M. Bühler, D. E. Koditschek, and P.J. Kindlmann. A family of robot control strategies for intermittent dynamical environments. In *Proc. IEEE International Conference on Robotics and Automation*, pages 1996–2002, Arizona, May 1989.
- [12] M. Bühler, D. E. Koditschek, and P.J. Kindlmann. A family of robot control strategies for intermittent dynamical environments. (to appear) *IEEE Control Systems Magazine*, 1990.
- [13] Martin Bühler. *Robotic Tasks with Intermittent Dynamics*. PhD thesis, Yale University, New Haven, CT, May 1990.
- [14] Robert L. Devaney. *Introduction to Chaotic Dynamical Systems*. Addison Wesley, Reading, MA, 1987.
- [15] Michael A. Erdmann. On motion planning with uncertainty. Technical Report AI-TR-810 (MS Thesis), MIT AI Lab, Aug 1984.
- [16] John Guckenheimer. Sensitive dependence to initial conditions for one dimensional maps. *Communications in Mathematical Physics*, (70):133–160, 1979.
- [17] Morris W. Hirsch and Stephen Smale. *Differential Equations, Dynamical Systems, and Linear Algebra*. Academic Press, Inc., Orlando, Fla., 1974.
- [18] R.S. Hartenberg J.J. Uicker, J. Denavit. An iterative method for displacement analysis of spatial mechanisms. *J. Applied Math.*, 31:309–314, June 1964.
- [19] O. Khatib and J.-F. Le Maitre. Dynamic control of manipulators operating in a complex environment. In *Proceedings Third International CISM-IFTOMM Symposium*, pages 267–282, Udine, Italy, Sep 1978.
- [20] Daniel E. Koditschek. Exact robot navigation by means of potential functions: Some topological considerations. In *IEEE International Conference on Robotics and Automation*, pages 1–6, Raleigh, NC, Mar 1987.
- [21] Daniel E. Koditschek. The Application of Total Energy as a Lyapunov Function for Mechanical Control Systems. In J. Marsden, Krishnaprasad, and J. Simo, editors, *Control Theory and Multibody Systems*, volume 97, pages 131–158. AMS Series in Contemporary Mathematics, 1989.
- [22] Daniel E. Koditschek and Elon Rimon. Robot navigation functions on manifolds with boundary. *Advances in Applied Mathematics*, (to appear).
- [23] Tomás Lozano-Perez and M. A. Wesley. An algorithm for planning collision free paths among polyhedral obstacles. *Commun. ACM*, 22(10), 1979.
- [24] Matthew T. Mason. Mechanics and planning of manipulator pushing operations. *International Journal of Robotics Research*, 5(3):53–71, Fall 1986.
- [25] M.T. Mason. Compliance and force control for computer controlled manipulators. *IEEE Trans. Syst. Man and Cybernetics*, SMC-11(6):418–432, 1981.
- [26] Tad McGeer. Passive dynamic walking. *The International Journal of Robotics Research*, 9(2):62–82, 1990.
- [27] Fumio Miyazaki and S. Arimoto. Sensory feedback based on the artificial potential for robots. In *Proceedings 9th IFAC*, Budapest, Hungary, 1984.
- [28] V. V. Pavlov and A. N. Voronin. The method of potential functions for coding constraints of the external space in an intelligent mobile robot. *Soviet Automatic Control*, (6), 1984.
- [29] Marc H. Raibert. *Legged Robots That Balance*. MIT Press, Cambridge, MA, 1986.
- [30] E. Rimon and D. E. Koditschek. Exact robot navigation using cost functions: The case of spherical boundaries in E^n . In *IEEE International Conference on Robotics and Automation*, pages 1791–1796, Philadelphia, PA, Apr 1988.
- [31] E. Rimon and D. E. Koditschek. The construction of analytic diffeomorphisms for exact robot navigation on star worlds. (to appear) *Transactions of the American Mathematical Society*, 1989.
- [32] E. Rimon and D. E. Koditschek. Exact robot navigation in topologically simple but geometrically complicated environments. In *Proc. IEEE International Conference on Robotics and Automation*, page (to appear), Cincinnati, OH, May 1990.
- [33] Elon F. Rimon. *Exact Robot Navigation Using Artificial Potential Functions*. PhD thesis, Yale University, New Haven, CT, Dec 1990.
- [34] Bernard Roth and Madhusudan Raguvan. Kinematic analysis of the 6r manipulator of general geometry. In *Proceedings Fifth International Symposium of Robotics Research*, pages 314–320, Tokyo, Japan, 1989.
- [35] Jacob T. Schwartz and Micha Sharir. On the “Piano Movers” problem i. the case of a two-dimensional rigid polygonal body moving amidst polygonal barriers. Technical Report 39, N.Y.U. Courant Institute Department of Computer Science, New York, 1981.
- [36] Jacob T. Schwartz and Micha Sharir. On the “Piano Movers” problem. II. general techniques for computing topological properties of real algebraic manifolds. *Advances in Applied Mathematics*, 4:298–351, 1983.
- [37] David Singer. Stable orbits and bifurcations of maps of the interval. *SIAM J. Applied Mathematics*, 35(2):260–267, Sep 1978.
- [38] Sir W. Thompson and P. G. Tait. *Treatise on Natural Philosophy*. University of Cambridge Press, 1886, Cambridge.
- [39] D. E. Whitney. Force feedback control of manipulator fine motions. *ASME J. Dyn. Syst.*, 98:91–97, June 1977.