Progress in spatial robot juggling

Alfred A. Rizzi * Daniel E. Koditschek †

*Yale University,

†University of Pennsylvania, kod@seas.upenn.edu

Proceedings of the 1992 IEEE
International Conference on Robotics and Automation
Nice, France - May 1992

Progress in Spatial Robot Juggling

A. A. Rizzi and D. E. Koditschek *
Center for Systems Science, Department of Electrical Engineering
Yale University, New Haven, CT 06520-1968

Abstract

We review our progress to date in eliciting dynamically dexterous behaviors from a three degree of freedom direct drive robot manipulator whose real-time stereo cameras provide 60 Hz sampled images of multiple freely falling bodies in highly structured lighting conditions. At present, the robot is capable of forcing a single ping-pong ball into a specified steady state (near) periodic vertical motion by repeated controlled impacts with a rigid paddle. The robot sustains the steady state behavior over long periods (typically thousands and thousands of impacts) and is capable of recovering from significant unexpected adversarial perturbations of the ball's flight phase. Gain tuning experiments corroborate our contention that the stability mechanism underlying the robot's reliability can be attributed to the same nonlinear dynamics responsible for analogous behavior in a one degree of freedom forebear. We are presently extending an algorithm for simultaneously juggling two bodies developed in that earlier work to the three dimensional case.

1 Introduction

This paper describes our progress in extending certain notions of how to achieve dynamical dexterity originally developed in the context of a one degree of freedom robot [8]. From the point of view of manipulation, the work suggests that good performance can be achieved by recourse to simple but appropriate contact models (we employ the naive coefficient of restitution as a means of describing the passage between free flight and ball-paddle impact) when effectively corrected by feedback. This would be fortunate if valid in any generality since the adoption of such facile descriptions results in mathematically tractable systems of equations whose stability properties have begun to yield to analysis [12, 6]. From the point of view of coordination, we seek increasingly taxing problems by which to test our assertion that a "geometric" mode of programming (feedback laws) can simultaneously offer an effective means of expression (representation of what we want a machine to do) along with a correct means of implementation (a mapping of sensor-readings to torque-profiles that produces the desired behavior). The intuitively generated extensions we present here of the single body "mirror law" to the two body case have as yet no better claim to analytical origins than any old computer program. But, by the same token, their generation has been no more arcane than writing code in any new computer language. Moreover, there is strong reason to hope that analyzing their effect will be simpler. Finally, from the point of view of sensor management, we have now reached an interesting stage of our work where it seems that previously viable algorithms may not avail in the absence of a more comprehensive approach to controlling our robot's state of attention.

The paper is organized as follows. The next section describes the pieces of our experimental apparatus along with the models we have adopted in thinking about and programming its operation. Section 3 presents a generalization to the present case of the mirror laws developed in our previous one degree of freedom planar juggler studies [8, 12]. Section 4 details the results to date of our empirical work.

*This work has been supported in part by the Superior Electric Corporation, SGS Thomson-INMOS Corporation and the National Science Foundation under a Presidential Young Investigator Award held by the second author.

2 Juggling Apparatus

This section describes the constituent pieces of our juggling machine. The system, pictured in Figure 1, consists of three major components: an environment (the ball); the robot; and an environmental sensor (the vision system). We now describe in fairly specific terms the hardware underlying each component and propose a (necessarily simplified) mathematical model in each case that describes its properties in isolation.

2.1 Environment: Striking a Ball in Flight

The two properties of the ball relevant to juggling are its flight dynamics (behavior while away from the paddle), and its impact dynamics (how it interacts with the paddle/robot). For simplicity we have chosen to model the ball's flight dynamics as a point mass under the influence of gravity. This gives rise to the flight model

$$\ddot{b} = \tilde{a} \tag{1}$$

where $b \in \mathcal{B} = \mathbb{R}^3$, and $\tilde{a} = (0, 0, -\gamma)^T$ is the acceleration vector experienced by the ball due to gravity. A position-time-sampled measurement of this system will be described by the discrete dynamics,

$$\begin{aligned} w_{j+1} &= F^s(w_j) \triangleq A_s w_j + a_s \\ b_j &= C w_j \end{aligned}$$
$$A_s \triangleq \begin{bmatrix} I & sI \\ 0 & I \end{bmatrix}; \quad a_s \triangleq \begin{bmatrix} \frac{1}{2}s^2\tilde{a} \\ s\tilde{a} \end{bmatrix}; \quad C = [I, 0], \tag{2}$$

0-8186-2720-4/92 $3.00 ©1992 IEEE

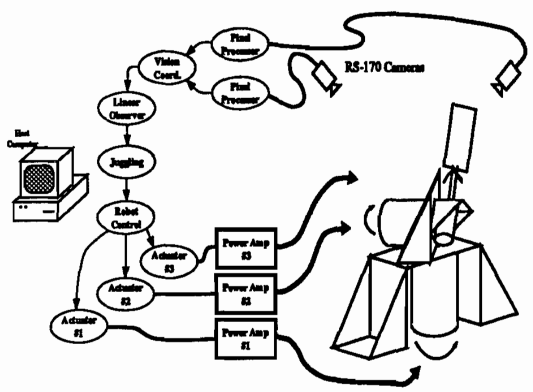

Figure 1: The Yale Spatial Juggler

where s denotes the sampling period, and $w_j \in T\mathcal{B}$.

Suppose a ball with trajectory $b(t)$ collides with the paddle in robot configuration $q \in \mathcal{Q}$ at some point, p on the paddle which has a linear velocity v. Letting $\mathcal{T} \triangleq \mathcal{B} \times \mathcal{Q}$ denote the total configuration space of the problem, we seek a description of how the ball's phase, $(b, \dot{b}) \in T\mathcal{B}$, is changed by the robot's phase, $(q, \dot{q}) \in T\mathcal{Q}$, at an impact.

As in [8, 12, 14] we will assume that the components of the ball's velocity tangent to the paddle at instant of contact are unchanged, while the normal component is governed by the simplistic (but standard [17]) coefficient of restitution law. For some $\alpha \in [0, 1]$ this impact model can be expressed as $(\dot{b}'_n - v'_n) = -\alpha(\dot{b}_n - v_n)$, where $\dot{b}'_n$ and v'_n denote the normal components of the ball and paddle velocities immediately after impact, while $\dot{b}_n$ and v_n are the velocities prior to impact. Assuming that the paddle is much more massive than the ball (or that the robot has large torques at its disposal), we conclude that the velocity of the paddle will remain constant throughout the impact ($v' = v$). It then follows that the coefficient of restitution law can be re-written as $\dot{b}'_n = \dot{b}_n + (1 + \alpha)(v_n - \dot{b}_n)$. and, hence,

$$\dot{b}' = \dot{b} + (1 + \alpha) n n^{\mathrm{T}} (v - \dot{b}), \qquad (3)$$

where n denotes the unit normal vector to the paddle.

2.2 Kinematic and Impact Models

At the heart of the juggling system is a three degree of freedom robot — the Bühgler Arm[1] — equipped at its end effector with a paddle. There are two primary aspects of the robot relevant to the juggling task, the kinematic properties associated with making contact with the environment, and the manner in which the robot can interact with the environment.

The robot kinematics relevant to the task of batting a ball relates the machine's configuration to the normal vector at a point on its paddle. In order to represent this formally we parameterize the paddle's surface geometry. Let $\bar{p}$ represent (in homogeneous coordinates) a planar transformation taking points in the unit box, $\mathcal{S} \triangleq [0, 1] \times [0, 1]$ diffeomorphically onto the paddle's (finite) surface area expressed with respect to the gripper frame, $\mathcal{F}_g$. Associated with each point on the paddle's surface, $\bar{p}(s)$, is the unit normal, $\bar{n}(s)$, again the homogeneous coordinate representation of the vector with respect to $\mathcal{F}_g$. Denote by $H(q)$ the robot's forward kinematic map taking a configuration, $q \in \mathcal{Q}$, to the homogeneous matrix representation of the gripper frame with respect to the base. The world frame representation of any paddle normal at a point is thus specified by the extended forward kinematic map,

$$\begin{gathered} G : \tilde{\mathcal{Q}} \rightarrow \mathcal{N}(3) \; : \tilde{q} \mapsto [n(\tilde{q}), p(\tilde{q})] = H(q)\,[\bar{n}(s), \bar{p}(s)] \\ \tilde{q} = (q, s) \in \tilde{\mathcal{Q}} \triangleq \mathcal{Q} \times \mathcal{S}. \end{gathered} \qquad (4)$$

Lying in the total configuration space is the contact submanifold, $\mathcal{C}$, — the set of ball/robot configurations where the ball is in contact with the paddle — given by

$$\mathcal{C} \triangleq \{(b, q) \in \mathcal{T} : \exists s \in \mathcal{S}, b = p(q, s)\},$$

which is evidently the only place that the normal appearing in (3) becomes relevant. Since $\bar{p}$ is one-to-one by assumption there is a map $s_c : \mathcal{C} \rightarrow \mathcal{S}$ such that $b = p(q, s_c(b, q))$.

For the Bühgler Arm, we choose a gripper frame, $\mathcal{F}_g$, located at the base of the paddle whose x-axis is aligned with the paddle's normal and whose z-axis is directed along the paddle's major axis, thus $\bar{n} = [1 \;\; 0 \;\; 0 \;\; 0]^T$ and $\bar{p} = [d_g \;\; s_1 \;\; s_2 \;\; 1]^T$ (for reasons to be made clear below we artificially set $s_1 = 0$). The frame transformation, $H(q)$, is developed in [15], and yields the extended kinematic map given by

$$\begin{aligned} G(q, s) &= [n(\tilde{q}), p(\tilde{q})] \qquad (5) \\ n(\tilde{q}) &= \begin{bmatrix} C_1C_2C_3 - S_1S_3 \\ C_2C_3S_1 + C_1S_3 \\ -C_3S_2 \\ 0 \end{bmatrix} \\ p(\tilde{q}) &= \begin{bmatrix} -(S_1 d_2) + (C_1C_2C_3 - S_1S_3)\,d_g + C_1S_2 s_2 \\ C_1 d_2 + (C_2C_3S_1 + C_1S_3)\,d_g + S_1S_2 s_2 \\ -(C_3S_2 d_g) + C_2 s_2 \\ 1 \end{bmatrix}. \end{aligned}$$

Finally the inverse kinematic image of a point b may be readily computed as

$$p^{-1}(b) = \begin{bmatrix} \operatorname{atan}(\frac{-b_2}{-b_1}) + \operatorname{asin}(\frac{\sin(q_3)d_2}{\sqrt{b_1^2+b_2^2}}) \\ -\frac{\pi}{2} \;\; + \;\; \operatorname{atan}(\frac{b_3}{\sqrt{b_1^2+b_2^2-\sin(q_3)d_2^2}}) - \\ \operatorname{asin}(\frac{\cos(q_3)d_g}{\sqrt{b^Tb-\sin^2(q_3)d_2^2}}) \\ q_3 \\ 0 \\ \sqrt{b^Tb - \sin^2(q_3)d_2^2 - \cos^2(q_3)d_g^2} \end{bmatrix}, \qquad (6)$$

with the freely chosen parameter, q_3, describing the one dimensional set of robot configurations capable of reaching the point b. Having simplified the kinematics via the artificial joint constraint, $s_1 \equiv 0$, the paddle contact map, s_c, may simply be read off the inverse kinematics function,

At the instant of impact with a ball the linear velocity of the hit point due the robot's motion may be written

[1] Pronounced *byōōg'–ler*.

explicitly as

$$v = \sum_{i=1}^{dim Q} \dot{q}_i D_{q_i} H\,(q)\bar{p}(s) = D_q p\, \dot{q} = Dp\, \Pi_Q \dot{q} \qquad \Pi_Q \triangleq [D\pi_Q]^{\mathrm{T}} \tag{7}$$

Notice that the appearance of Π_Q eliminates the velocity contribution due to the ball's motion tangential to the paddle surface.

Combining this with the paddle contact map s_c we may rewrite the impact event (3) in terms of a "collision map" $\mathrm{c} : T\mathcal{C} \rightarrow T\mathcal{B}$, as

$$\begin{aligned} \dot{b}' &= \dot{b} + \mathrm{c}(b, \dot{b}, q, \dot{q}) \\ \mathrm{c}(b, \dot{b}, q, \dot{q}) &\triangleq -(1+\alpha) n(q, s_c) n^{\mathrm{T}}(q, s_c) \left(\dot{b} - Dp\, \Pi_Q \dot{q}\right). \end{aligned} \tag{8}$$

Note that the vector direction attained by c is governed entirely by the choice of some the paddle normal. This will serve the role of input in an abstracted model of impact events to be developed below (9).

2.3 Sensors: A Field Rate Stereo Vision System

Two RS-170 CCD television cameras constitute the "eyes" of the juggling system. In order to make the vision task tractable we have simplified the environment the vision system must interpret. The "world" as seen by the cameras contains only one or more white balls against a black background. The CYCLOPS vision system, described in [14, 9], allows the straightforward integration of these cameras into the larger system.

Following Andersson's experience in real-time visual servoing [2] we employ the result of a first order moment computation applied to a small window of a threshold-sampled (that is, binary valued) image of each camera's output. Thresholding, of course, necessitates a visually structured environment, and we presently illuminate white ping-pong balls with halogen lamps while putting black matte cloth cowling on the robot, floor, and curtaining off any background scene.

In practice it is necessary to associate a signal processing system with the sensor to facilitate interpretation of the data. For the vision system in use here, sensor interpretation consists of estimating the ball's position and velocity, correcting for the latency of the vision system, and improving the data rate out of the sensor system – the 60 Hz of the vision system is far below the bandwidth of the robot control system.

Given reports of the ball's position from a triangulator it is straightforward to build a linear observer for the full state — positions and velocities — since (A_s, C) in (2) is an observable pair for $s \neq 0$. In point of fact, it is not the ball's position, b_n, which is input to the observer, but the result of a series of computations applied to the cameras' images, and this "detail" comprises the chief source of difficulty in building cartesian sensors of this nature.

2.4 Trajectory Tracking via Inverse Dynamics

The manipulator's kinematics (5) gives rise to enormously complicated nonlinear dynamics through application of the familiar Euler-Lagrange operator to the kinetic energy function [1, 3]. In our original implementation of the spatial vertical one-juggle [14] we obtained sufficiently good manipulator response by recourse to simple decoupled proportional-derivative feedback around the desired joint trajectories described in the sequel. The new vertical two-juggle task taxes the manipulator's capabilities sufficiently that we have shifted to an inverse dynamics scheme that matches exactly our model of the robot's Lagrangian dynamics. In order to obtain the dynamical parameters required for matching we run an adaptive version of this algorithm [19] as a part of the calibration procedure for the machine.

2.5 Controller: A Flexibly Reconfigurable Computational Network

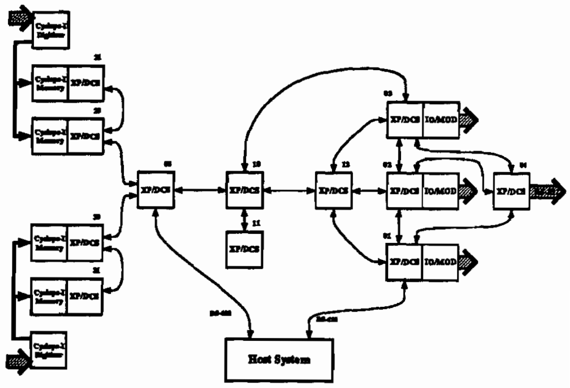

Figure 2: The Spatial Juggling System: Network Diagram.

A block diagram for the interconnection of signals modeled above can be seen on the left side of Figure 1. It should be clear that with the exception of the signal processing and motor operation blocks of this diagram, we are faced with the need for a very large amount of computation at varying data rates.

All of the growing number of experimental projects within the the Yale University Robotics Laboratory are controlled by widely varying sized networks of Transputers, produced by the INMOS division of SGS-Thomson [13, 10, 18]. The recourse to parallel computation considerably boosts the processing power per unit cost that we can bring to bear on any laboratory application. At the same time the serial communication links have facilitated quick network development and modification. Figure 2 depicts the network used to support the particular apparatus described in this paper [16].

3 Juggling Algorithms

This section describes how the juggling analysis and control methodology originally introduced for the planar system [8] may be extended in a straightforward manner to the present apparatus. Introducing the "environmental control system," a higher level discrete representation of the robot's effect upon the ball's motion, clarifies how to encode an elementary but abstract task, the "vertical one juggle," as a stable equilibrium state of an appropriate dynamical system. A simple computation reveals that every achievable vertical one juggle can be made a fixed point, and conversely, the only fixed points of the environmental control system are those that encode a vertical one juggle, confirming the clear intuition that the task is logically achievable. Leapfrogging the intermediate linearized analysis of our planar work [4], we pass immediately to a "commented" presentation of the mirror law, a nonlinear function from the phase space of the body to the phase space of the robot that generates a reference trajectory — a time varying profile of desired robot states — as the ongoing time history of the ball's position and velocity is fed through it. Finally, the mirror law, now considered as a behavioral primitive, is used in conjunction with certain analytic functions that intuitively implement our notions of "if-then-else" within this "geometric programming framework," to develop a juggling strategy for keeping two balls aloft simultaneously.

It must be emphasized that the functions we present here comprise at once a mathematical description of our algorithm and its actual implementation. Implementating "geometric programs" of this type amounts to merely placing the particular transformation law — in the present case, (14) or (15) — in the juggling block of the data flow path depicted in the left side of Figure 1. One immediate practical benefit of this arrangement is the availability of very powerful high level development environments in the form of commercial symbolic manipulation packages. In practice, we craft these functions in Mathematica on a SPARCstation and use the automatically generated C code on the target controller.

3.1 Task Encoding

Denote by $\mathcal{V}$ the robot's choices of impact normal velocity for each workspace location. Suppose that the robot strikes the ball in state $w_j = (b_j, \dot{b}_j)$ at time s with a velocity $v_j = (q, \dot{q}) \in \mathcal{V}$ and allows the ball to fly freely until time $s + t_j$. According to (8), composition with time of flight (2) yields the "environmental control system"

$$w_{j+1} = f(w_j, v_j, t_j) \triangleq A_{t_j} w_j + a_{t_j} + \begin{bmatrix} t_j \mathbf{c}(w_j, v_j) \\ \mathbf{c}(w_j, v_j) \end{bmatrix}, \quad (9)$$

that we will now be concerned with as a controlled system. with control inputs in $\mathcal{V} \times \mathrm{IR}$ (v_j and t_j).

Probably the simplest systematic behavior of this system imaginable (beyond the ball at rest on the paddle), is a periodic vertical motion of the ball. In particular, we want to be able to specify an arbitrary "apex" point, and from arbitrary initial conditions, force the ball to attain a periodic trajectory which passes through that apex point. This corresponds exactly to the choice of a fixed point, w^*, in (9), of the form

$$w^* = \begin{bmatrix} b^* \\ \dot{b}^* \end{bmatrix}; \quad b^* \in \mathrm{IR}^3; \quad \dot{b}^* = \begin{bmatrix} 0 \\ 0 \\ \nu \end{bmatrix}; \quad \nu \in \mathrm{IR}^1, \quad (10)$$

denoting a ball state-at-impact occurring at a specified location, with a velocity which implies purely vertical motion and whose magnitude is sufficient to bring it to a prespecified height during free flight. Denote this four degree of freedom set of vertical one-juggles by the symbol $\mathcal{J}$.

The question remains as to which tasks in $\mathcal{J}$ can be achieved by the robot's actions (i.e. which elements of $\mathcal{J}$ can be made fixed points of (9)). Analysis of the fixed point conditions [15] similar to that of [8] shows that only those elements of $\mathcal{J}$ satisfying the condition

$$b^* \in p(\mathcal{Q}^*); \quad \mathcal{Q}^* \triangleq \{q \in \mathcal{Q} : \cos(q_3)\sin(q_2) = -1\},$$

will be fixable. In particular, $\mathcal{Q}^*$ corresponds to the paddle being positioned parallel to the floor, and thus $p(\mathcal{Q}^*)$ is an annulus above the floor, as is intuitively expected.

3.2 Controlling the Vertical One-Juggle via a Mirror Law

Say that the abstract feedback law for (9), $g : \mathcal{W} \to \mathcal{V} \times \mathrm{IR}$, is a verticle one-juggle strategy if it induces a closed loop system,

$$f_g(w) = f(w, g(w)), \quad (11)$$

for which $w^* \in \mathcal{J}$ is an asymptotically stable fixed point. For our original planar machine [4] it was shown that the linearization of the analogous system to (9) was controllable around every vertical one juggle task. A similar analysis has not yet been completed for the Bühgler Arm, although a similar result is expected.

In [8] a novel juggling strategy was proposed that implicitly realized an effective discrete feedback policy, g, by requiring the robot to track a distorted reflection of the ball's continuous trajectory. This policy, the "mirror law," may be represented as a map $m : T\mathcal{B} \to \mathcal{Q}$, so that the robot's reference trajectory is determined by

$$q(t) = m(w(t)).$$

The juggling algorithm used in the present work is a direct extension of this "mirror" law to the spatial juggling problem. In particular begin by using (6) to define the the joint space position of the ball

$$\begin{bmatrix} \phi_b \\ \theta_b \\ \psi_b \\ s_b \end{bmatrix} \triangleq p^{-1}(b). \quad (12)$$

We now seek to express formulaically a robot strategy that causes the paddle to respond to the motions of the ball in four ways:

(i) $q_{d1} = \phi_b$ causes the paddle to track under the ball at all times.

(ii) The paddle "mirrors" the vertical motion of the ball through the action of θ_b on q_{d2} as expressed by the original planar mirror law [8].

(iii) Radial motion of the ball causes the paddle to raise and lower, resulting in the normal being adjusted to correct for radial deviation in the ball position.

(iv) Lateral motion of the ball causes the paddle to roll, again adjusting the normal so as to correct for lateral position errors.

To this end, define the ball's *vertical energy* and *radial distance* as

$$\eta \triangleq \gamma b_z + \frac{1}{2}\dot{b}_z^2 \quad \text{and,} \quad \rho_b \triangleq \sin(\theta_b) s_b \tag{13}$$

respectively. The complete mirror law combines these two measures with a set point description ($\bar{\eta}$, $\bar{\rho}$, and $\bar{\phi}$) to form the function

$$\mathrm{m(w)} \triangleq \begin{bmatrix} \underbrace{-\frac{\pi}{2} - (\kappa_0 + \kappa_1(\eta - \bar{\eta}))\left(\theta_b + \frac{\pi}{2}\right)}_{\text{(ii)}} + \overbrace{\phi_b}^{\text{(i)}} \\ \underbrace{\kappa_{00}(\rho_b - \bar{\rho}_b) + \kappa_{01}\dot{\rho}_b}_{\text{(iii)}} \\ \underbrace{\kappa_{10}(\phi_b - \bar{\phi}_b) + \kappa_{11}\dot{\phi}_b}_{\text{(iv)}} \end{bmatrix}. \tag{14}$$

3.3 The Two-Juggle

A two-juggle task requires that the robot perform two simultaneous one-juggle tasks with two independent balls separated in both space and time. Separation in space avoids ball-ball collisions, which are not currently part of the environmental model, while temporal separation (meaning that the two balls should not fall simultaneously) is necessary to ensure that the machine is capable of striking one ball and moving into position under the second, prior to the first falling to the floor. Apparently there is an obstacle present in the phase space of the system which we are attempting to "avoid". Thus the juggling algorithm must be able to control the phase relationship between the two balls in addition to the three new variables associated with the position and energy of the additional ball.

Individual mirror laws for the two balls, constructed as in [7], are combined to form the overall two-juggle law by the use of a scalar valued analytic switch $s \in [0, 1]$,

$$\begin{aligned} m_{II}(w_0, w_1) &= s(w_0, w_1) m_0(w_0, w_1) + \\ &\quad (1 - s(w_0, w_1))\, m_1(w_1, w_0). \end{aligned} \tag{15}$$

The function s encodes the mixture between the need to juggle ball 0 (follow m_0) or ball 1 (follow m_1). An "urgency" function is defined for each ball by

$$\begin{aligned} \sigma(w) = \; & \tfrac{1}{2}\Big(1 - \cos\Big(2\,\mathrm{atan}(\kappa_c\left(\kappa_b + \tan(\tfrac{\epsilon\pi}{2})\right) + \\ & \kappa_a\left(\kappa_b + \tan(\tfrac{\epsilon\pi}{2})\right)^3)\Big)\Big), \end{aligned}$$

where $\epsilon(w) \triangleq -\frac{\dot{b}_3}{\sqrt{2\eta}}$ is the *phase* of the ball. Finally s is then given by

$$s = \frac{1 - \sigma(w_1)}{2 - \sigma(w_0) - \sigma(w_1)}. \tag{16}$$

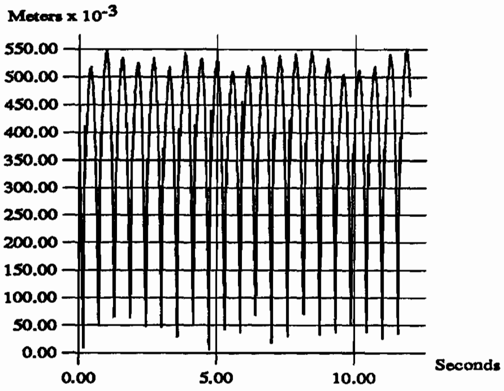

Figure 3: One-Juggle ball trajectory: Height vs. Time.

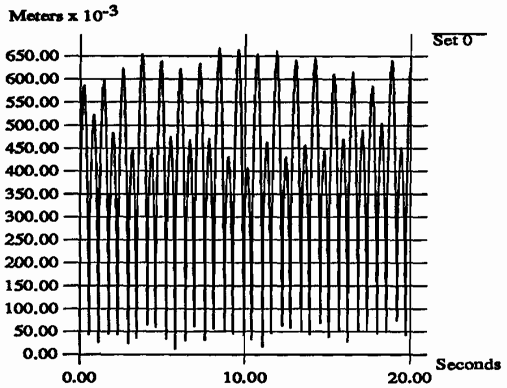

Figure 4: Period two One-Juggle ball trajectory: Height vs. Time.

The motivation for this implementation is as follows. ϵ varries smoothly from -1 immediately after impact, to 0 at the balls apex, to 1 immediately prior to impact. σ then describes the *urgency* of the ball (being near 1 when the ball is near impact, and 0 as it rises to its apex). Finally s combines these two *urgencies* by smoothly mapping the unit box onto $[0, 1]$ so $s = 0$ when $\sigma_1 = 1$, and $s = 1$ when $\sigma_0 = 1$.

4 Present Status

We have succeeded in implementing the one-juggle task as defined in Section 3.1 on the Bühgler arm. The overall performance of the constituent pieces of the system – vision module, juggling algorithm, and robot controller – have each been outstanding, allowing for juggling behavior that is gratifyingly similar to the impressive robustness and reliability of the planar juggling system. We typically record thousands of impacts (hours of juggling) before random system imperfections (electrical noise, paddle inconsistencies) result in failure [14]. Figure 3 depicts the ball's height over time over a short segment of a typical run, and demon-

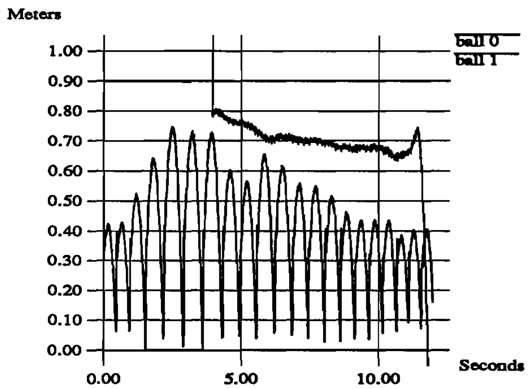

Figure 5: Two-Juggle ball trajectories: Height vs. Time.

strates that the machine has been capable of controlling the height of the ball trajectories to within 5cm of the desired height of 55cm.

Analysis of our planar juggler [6] revealed that the strong stability and robustness properties of the machine could be attributed to the Singer-Guckenheimer unimodal return map [11, 12]. We have not yet undertaken an analogous formal study of the closed loop return map (11) induced by the spatial mirror law (14) but conjecture that the same stability mechanism will be found here too. In advance of this analysis, we note in passing our ability to generate the bifurcation to a period two orbit in (11) predicted by the Singer-Guckenheimer theory as depicted in Figure 4.

To date we have not achieved a working two-juggle. The usual sequence of events follows the pattern:

(i) Introduce the initial ball which the machine juggles well.

(ii) Introduce the second ball while the machine continues to juggle the first.

(iii) Release the second ball. This generally results in one of the balls being missed, while the other is "captured" and batted into its specified steady state one-juggle limit cycle.

As an example, Figure 5 shows the height of both balls during a typical two-juggle attempt. In this case the second ball was missed immediately after release, while the machine continued to juggle the first ball. Extremely poor release of the second ball (closely in phase with the first ball) has resulted in both balls dropping. Conversely, extremely favorable release of the second ball (nearly out of phase) results in multiple impacts with both balls. In no case have we yet seen two-juggles of a quality at all reminiscent of that achieved in the planar case [5]. We expect that careful consideration od the robot's focus of attention — "sharpening" its knowledge of when and where either ball was batted — should substantially improve the juggling performance. We are currently exploring this class of improvments.

References

[1] Ralph Abraham and Jerrold E. Marsden. *Foundations of Mechanics*. Benjamin/Cummings, Reading, MA, 1978.

[2] R. L. Andersson. *A Robot Ping-Pong Player: Experiment in Real-Time Intelligent Control*. MIT, Cambridge,MA, 1988.

[3] Antal K. Bejczy. Robot arm dynamics and control. Technical Report 33-699, Jet Propulsion Laboratory, Pasadena, CA, 1974.

[4] 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.

[5] M. Bühler , D. E. Koditschek, and P.J. Kindlmann. Planning and Control of Robotic Juggling Tasks. In H. Miura and S. Arimoto, editors, *Fifth International Symposium on Robotics Research*, pages 321–332. MIT Press, 1990.

[6] M. Bühler and D. E. Koditschek. From stable to chaotic juggling. In *Proc. IEEE International Conference on Robotics and Automation*, pages 1976–1981, Cincinnati, OH, May 1990.

[7] M. Bühler, D. E. Koditschek, and P. J. Kindlmann. Planning and control of a juggling robot. *International Journal of Robotics Research*, (to appear), 1992 .

[8] M. Bühler, D. E. Koditschek, and P.J. Kindlmann. A family of robot control strategies for intermittent dynamical environments. *IEEE Control Systems Magazine*, 10:16–22, Feb 1990.

[9] M. Bühler, N. Vlamis, C. J. Taylor, and A. Ganz. The cyclops vision system. In *Proc. North American Transputer Users Group Meeting*, Salt Lake City, UT, APR 1989.

[10] M. Bühler, L. Whitcomb, F. Levin, and D. E. Koditschek. A new distributed real-time controller for robotics applications. In *Proc. 34th IEEE Computer Society International Conference — COMPCON*, pages 63–68, San Francisco, CA, USA, Feb 1989. IEEE Computer Society Press.

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

[12] D. E. Koditschek and M. Bühler. Analysis of a simplified hopping robot. *International Journal of Robotics Research*, 10(6), Dec 1991 .

[13] F. Levin, M. Bühler, L. Whitcomb, and D. E. Koditschek. Transputer computer juggles real-time robotics. *Electronic Systems Design*, 19(2):77–82, Feb 1989.

[14] Alfred Rizzi and Daniel E. Koditschek. Preliminary experiments in robot juggling. In *Proc. Int. Symp. on Experimental Robotics*, Toulouse, France, June 1991. MIT Press.

[15] Alfred A. Rizzi and Daniel E. Koditschek. Progress in spatial robot juggling. Technical Report 9113, Yale University, Center for Systems Science, New Haven, Connecticut, USA, October 1991.

[16] Alfred A. Rizzi, Louis L. Whitcomb, and D. E. Koditschek. Distributed real-time control of a spatial robot juggler. *IEEE Computer*, 25(5), May 1992.

[17] J. L. Synge and B. A. Griffith. *Principles of Mechanics*. McGraw Hill, London, 1959.

[18] L. L. Whitcomb and D. E. Koditschek. Robot control in a message passing environment. In *Proc. IEEE International Conference on Robotics and Automation*, pages 1198–1203, Cincinnati, OH, May 1990.

[19] Louis L. Whitcomb, Alfred Rizzi, and Daniel E. Koditschek. Comparative experiments with a new adaptive controller for robot arms. In *Proc. IEEE Int. Conf. Rob. and Aut.*, pages 2–7, Sacramento, CA, April 1991. IEEE Computer Society.