IEEE Conference on Robotics and Automation, Minneapolis, MN, April 1996, Vol. 4, pp. 2988-2993.

A Control Basis for Multilegged Walking*

Manfred Huber, Willard S. MacDonald, Roderic A. Grupen

Laboratory for Perceptual Robotics†
Department of Computer Science
University of Massachusetts, Amherst MA 01003

Abstract

This paper presents a distributed control approach to legged locomotion that constructs behavior on-line by activating combinations of reusable feedback control laws drawn from a control basis. Sequences of such controller activations result in flexible aperiodic step sequences based on local sensory information. Different tasks are achieved by varying the composition functions over the same basis controllers, rather than by geometric planning of leg placements or the design of new task-specific behaviors. In addition, the device-independent nature of the control basis allows its generalization not only over task domains, but also over different hardware platforms. To show the applicability of this approach, a control basis and two generic control gaits for four-legged walking are introduced and tested on an even terrain walking task in an unknown environment.

1 Introduction

In order to render a mobile robotic system autonomous it is important to make it capable of traversing various kinds of, possibly unknown, terrain. For this task, legged locomotion offers advantages over wheeled systems due to greater robustness and flexibility with respect to the terrain. It poses, however, additional control challenges through a higher number of degrees of freedom, explicit stability considerations, and an only indirect relationship between leg motions and the overall motion of the robot. Control, and especially gait planning for legged robots, has therefore received a lot of attention over the last 10 years. As most of the work in legged locomotion, this paper will focus on quasistatic walking, i.e. situations in which dynamic effects are negligible. A large body of work in this area has focused on geometric, periodic gaits for each aspect of walking [5, 13]. Search methods have been used to refine the actual footplacements, thus creating aperiodic gaits, in order to conform to contingencies posed by the terrain [7, 8, 10]. Typically, these gaits are computed off-line and can thus be quite sensitive to incomplete or imprecise terrain maps.

To confront uncertainties and model imprecision with increased flexibility and robustness, behavior-based architectures [1, 11] were introduced. In this bottom-up paradigm, global behavior is constructed on-line from combinations of reactive elemental behaviors. These systems are used to react to uncertainties in the execution of preplanned geometric leg sequences [7, 14], or to generate the actual foot placements on-line [1]. The procedural nature of the behaviors, however, can lead to extremely complex organizations of *ad hoc* behavioral elements which do not scale well and can not generalize across tasks and different robotic platforms.

In the approach presented here, a basis set of reusable, device independent feedback controllers which can be dynamically bound to specific system resources is used to construct reactive control gaits. Throughout execution, no geometric foot placements are generated, but rather control actions are derived on-line from a sequence of concurrent activations of a subset of the control basis. This results in the simultaneous control of foot placements and body stabilization. The arising step patterns are aperiodic and depend on local sensory information and the task objectives.

2 The Control Basis Approach

In the control scheme presented in this paper, behavior is constructed from a set of base controllers which represent generic control objectives. Careful

*This work was supported in part by NSF IRI-9116297, CDA-8922572 and IRI-9208920

†http://piglet.cs.umass.edu:4321

design of this control basis leads to a small set of control laws that span a broad class of tasks and are robust with respect to perturbations. As shown in Figure 1, control is derived on-line by associating input resources (sensors or sensor abstractions) and output resources (actuators) with elements of the control basis. These controllers are thereby activated concurrently according to a task-dependent composition policy in either an asynchronous fashion or under the "subject to" ("$\triangleleft$") constraint. The latter, similar to pseudoinverse control [15] where lower level controllers can only operate within the nullspace of the higher level controllers, restricts control actions of "subordinate" controllers to areas which do not counteract the control objectives of "dominant" controllers. Thus, the set of basis controllers need not be orthogonal with respect to the control objectives, but rather can be orthogonalized using the "$\triangleleft$" composition.

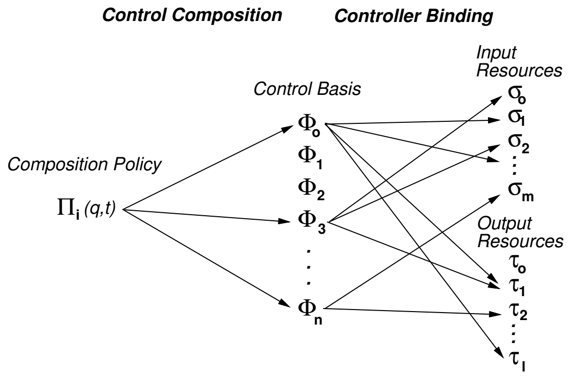

Figure 1: Control Composition

A complete control policy is then a sequence of concurrent control activations of the form described above with transitions caused by convergence events. Composition functions can thus be represented as finite state supervisors in a discrete event dynamic system (DEDS) framework [12]. Due to the predictable performance of the control basis, the supervisor can be modeled on a logical level with all sensory patterns interpreted within the elemental controllers. To accomplish complex tasks, the composition policy leads the system through a set of favorable equilibria to the goal. Different tasks are achieved by different composition functions over the *same* basis controllers, rather than by the design of new task-specific behaviors as in behavior-based approaches. In addition, the kinematics-independent nature of the underlying control basis allows the same set of controllers to be used on different robot platforms.

2.1 A Control Basis for Walking Tasks

To demonstrate the applicability of the approach described in the previous section to walking tasks, a control system was designed and applied to example tasks. The control basis used for walking consists of solutions to three generic robot control tasks, namely configuration space motion control, contact configuration control, and kinematic conditioning. Although a detailed description of the controllers is beyond the scope of this paper, this section will give a short overview of each along with a discussion of the composition mechanism.

Φ_0: Configuration space motion control.
Harmonic function path controllers [4] are used to generate robust, reactive, and collision-free motion through the configuration space of the robot. Formally, this approach minimizes collision probabilities for the robot system [3] by following the gradient of a harmonic potential computed over the configuration space of the robot.

Φ_1: Contact configuration control.
A contact configuration controller [2] is employed to move contacts based on the local geometry of the environment in order to minimize residual forces and moments acting on the center of mass of the platform and thus to stabilize the robot.

Φ_2: Kinematic conditioning.
The kinematic conditioning controller optimizes the posture of the articulated structure while it is engaged in an interaction with the world. This allows the robot to maintain favorable leg postures and thus to passively adjust to changes caused by other controllers.

Each instance of these controllers takes the form $\Phi_i \frac{\sigma}{\tau}$, where Φ_i is an element of the control basis, and superscript σ and subscript τ denote the sets of input and output resources, respectively, which are bound to this controller. In the most general terms, σ and τ are sensors and actuators associated with the legs or abstractions derived from the stream of sensor data such as the estimated location of the center of mass of the platform. Resolving individual degrees of freedom within a series kinematic chain is not necessary. Figure 2 shows the set of controllers (Φ_0, Φ_1, Φ_2) and possible robot resources (legs $0, 1, 2, 3$ and center of mass x, y, φ) used for the quadruped walking gaits and experiments presented in the next sections.

Task level control is then achieved by composing these control elements in a context-dependent way.

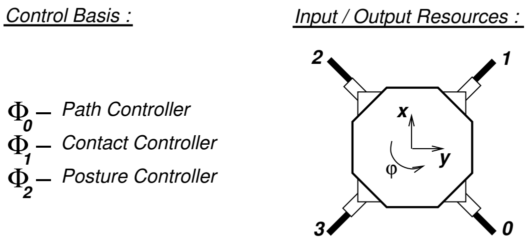

Figure 2: Control Gait Notation

Figures 3 and 4 present examples of the type of finite state supervisors employed by this work. The control policy at each point in time is shown inside each state as a set of asynchronous (";") and/or hierarchical ("$\lhd$") activations of controllers bound to specific system resources. Tasks are therefore not defined in terms of geometric foot placements but rather as stability, kinematic conditioning, and navigational objectives. Body stability and forward progress are established explicitly through the activation of corresponding controllers. Throughout execution, the safety of the composite system can thus be guaranteed within the controller capabilities and the sensory and motor limitations of the robot.

2.2 Control Gaits for Quadruped Walking

In order to allow the system to perform general walking, two major tasks were implemented in the form of control gaits, namely pure rotation and forward translation. The finite state control policy for the rotation task is shown in Figure 3.

Starting from an initial four-legged stance, this gait first shifts the center of mass (x, y) of the robot such as to form a stable stance on legs 0, 1, and 2, while the posture controller asynchronously adjusts the orientation φ of the body to condition all 4 legs. After this initial control state the control policy lets the robot system cycle through a succession of stable, three-legged stances, moving one leg at a time, while achieving the rotation through kinematic conditioning around the z-axis.

In order to allow spatial progress of the system, a second gait for general forward walking was implemented. Exhibiting a similar periodic structure as the rotation gait, the control policy for this gait is shown in Figure 4. In this gait the robot maintains its stability by cycling through a sequence of three and four-legged stances, moving the individual legs in the sequence $3, 2, 0, 1$. At the same time the control states attempt to optimize the kinematic configuration of the legs subject to the stability constraints imposed by the contact controller. As an example, $\left(\Phi_2 \frac{3}{\underline{3}} \lhd \Phi_1 \frac{0,1,3}{\underline{3}}\right)$ activates the contact controller, Φ_1, in order to establish a stable three-legged stance on legs $0, 1,$ and 3 by moving leg 3. Limited by the "subject to" constraint, the posture controller, Φ_2, then optimizes the posture of leg 3 within the range of stable stances.

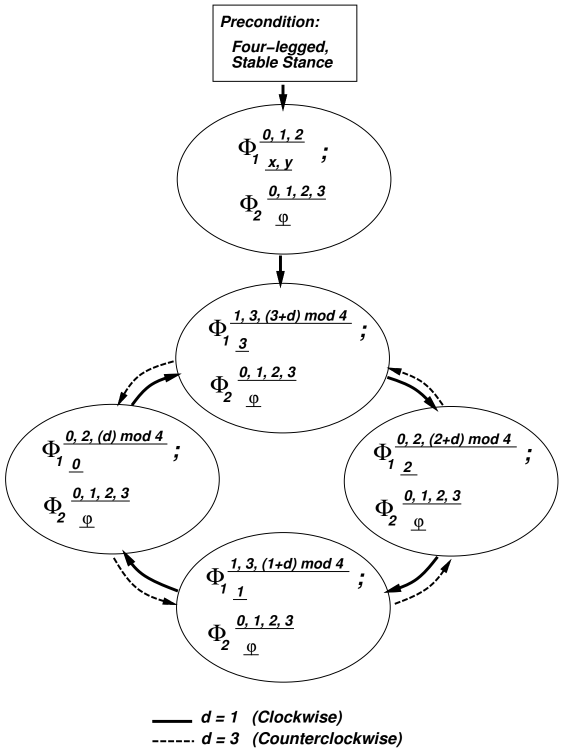

Figure 3: Control Gait for Rotating in Place

In addition to stabilizing the body, this gait also includes two states that activate instances of the path controller $\Phi_0 \frac{x,y}{\underline{x,y,\varphi}}$ subject to the stability constraints of the existing three-legged stance. This allows the gait to adjust the body location and orientation to conform with the direction of the current path.

In contrast to geometric gait planning where a different set of repetitive foot patterns is used to adjust to different walking directions, the control gaits used here continuously recompute footplacements based on the contingencies of the task and local sensory information. This can lead to significantly more flexible and reactive performance. The walking gait, for example, is capable of walking not only in a straight line but also at limited angles without changes in the heading. Moreover, it can execute path curvatures with instantaneous radii larger than approximately $0.4m$. In

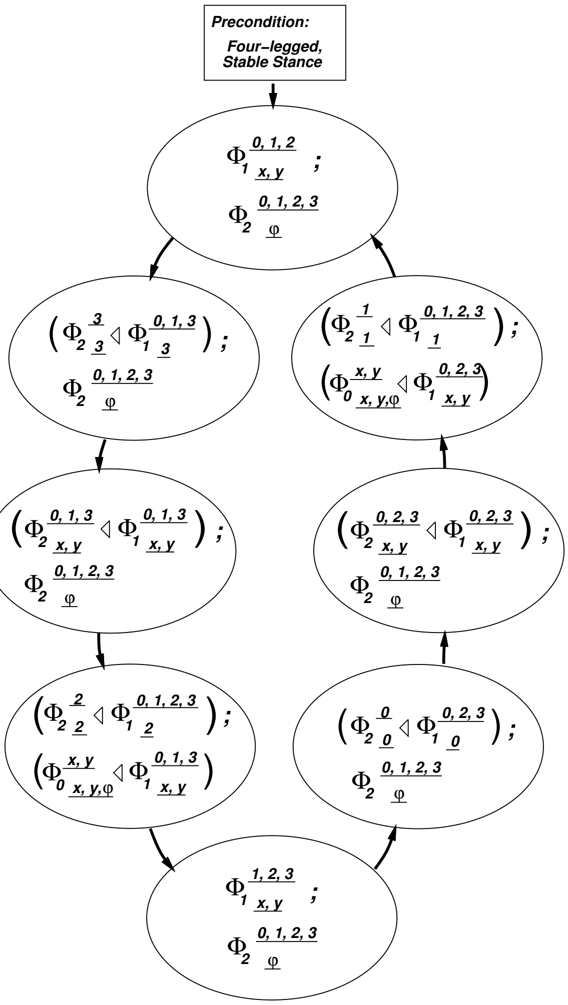

Figure 4: Control Gait for Forward Walking

geometric terms this strategy encompasses therefore aperiodic versions of discontinuous walking as well as crab and turning gaits [5], augmented by a reactive control component. In addition to this reactivity and the simplified gait switching, this control gait is also still largely device independent and thus generalizes well to other four-legged walking robots.

To show the applicability of these gaits and their performance, they were used on the four legged robot described in the next section and applied to an example task of walking in an unknown environment.

3 "Thing" - A Four-Legged Robot

The experimental platform shown in Figure 5 is a four-legged, 12 degree-of-freedom walking robot designed for autonomous locomotion over irregular terrain [9]. It stands $0.25m$ tall and weighs approximately $1.5kg$. The robot is equipped with five 8-bit microprocessors connected, in a serial-star configuration in addition to a serial tether to a workstation.

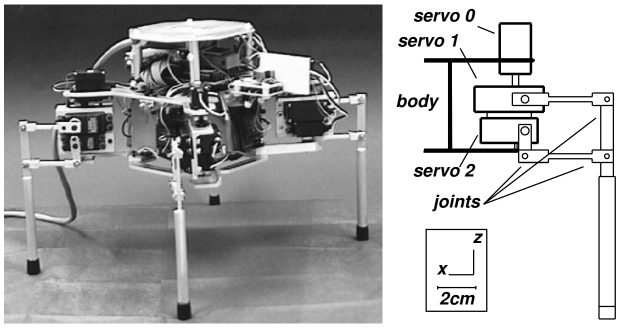

Figure 5: Quadruped Walking Robot "Thing" (left) and Leg Design (right)

The legs are four-bar linkage mechanisms with fork-and-shaft joints actuated by PWM controlled servos as shown in Figure 5. Servo 0 rotates about the *z-axis* and provides the swing motion of the leg while Servo 1 rotates about the *y-axis* (into the page), supporting the weight of the robot. A passive spring is utilized to counteract a portion of this weight. Servo 2 also rotates about the *y-axis* and drives the "knee" motion of the leg. The legs are made of milled aluminum and were designed for low cost, ease of construction, high efficiency, and maximum workspace.

The sensory apparatus consists of two infrared (IR) proximity sensors which provide a binary signal indicating the presence of obstacles within a range of $0.2m$. The sensor configuration allows to determine the existence of obstacles within three sectors in front of the robot. Odometry is maintained by accruing relative translations and rotations of the center-of-mass of the robot with respect to a known initial configuration.

4 Experiments

To demonstrate the capabilities of the composite controllers described above, the two control gaits were integrated using the switching policy shown in Figure 6, where $\Delta\varphi$ is the difference between the current and the desired heading given by the path controller.

Using this control scheme, the robot was placed in an unknown environment with the task of reaching a specified goal. Utilizing the IR sensors and odometry, the robot builds a map of the local environment as shown in the left column of Figure 7. This map,

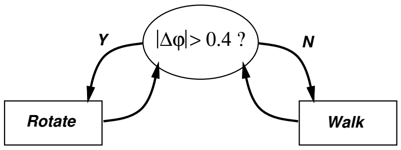

Figure 6: Gait Change Policy for Example Task

which models a $9m^2$ region is internally represented as a 32x32 cell grid and used by the path controller. When an obstacle is detected, the corresponding location is marked on the grid and dilated appropriately to reflect the geometry of the platform. All movements of the robot, including foot placements and global navigation, are then generated reactively on-line by the activated controllers in the control gait without the need for off-line planning. This allows the system to react very quickly to the detection of new obstacles without the need for frequent gait transitions.

Figure 7 shows the robot's state at six points during execution. Panels in the left column show the robot's internal map, and panels in the right column show the corresponding actual state of the robot and environment. The markers in the top panel of the robot's map represent the goal (on the left) and the starting point (on the right) and correspond to the two markers shown in the images in the right hand column. Since no obstacles other than the boundaries of the internal map have been mapped for the path controller, the robot initially heads directly from start to goal. When it encounters an obstacle, it is included in the map as shown in the second panel and the grid is relaxed to compute a new harmonic surface, providing a new desired heading for the path controller. Grid relaxation is repeated each time a new obstacle is found.

The gradient of the harmonic function influences the path of the robot in two ways: indirectly through the gait switching policy of Figure 6, and directly by means of the activated instances of the path controller in the forward walking gait. This allows the robot not only to turn away from newly discovered obstacles but also to follow the smoothly curved trajectories leading from the current location to the goal. For example when most of the obstacle barrier has been detected, the best path to the goal becomes the long arc around the entire barrier, which is the path finally taken as shown in the bottom panel.

Throughout the total walking time of 20 minutes the robot activated the rotation gait only 3 times, traveled approximately $5m$, and accrued a translational odometry error of approximately $0.2m$.

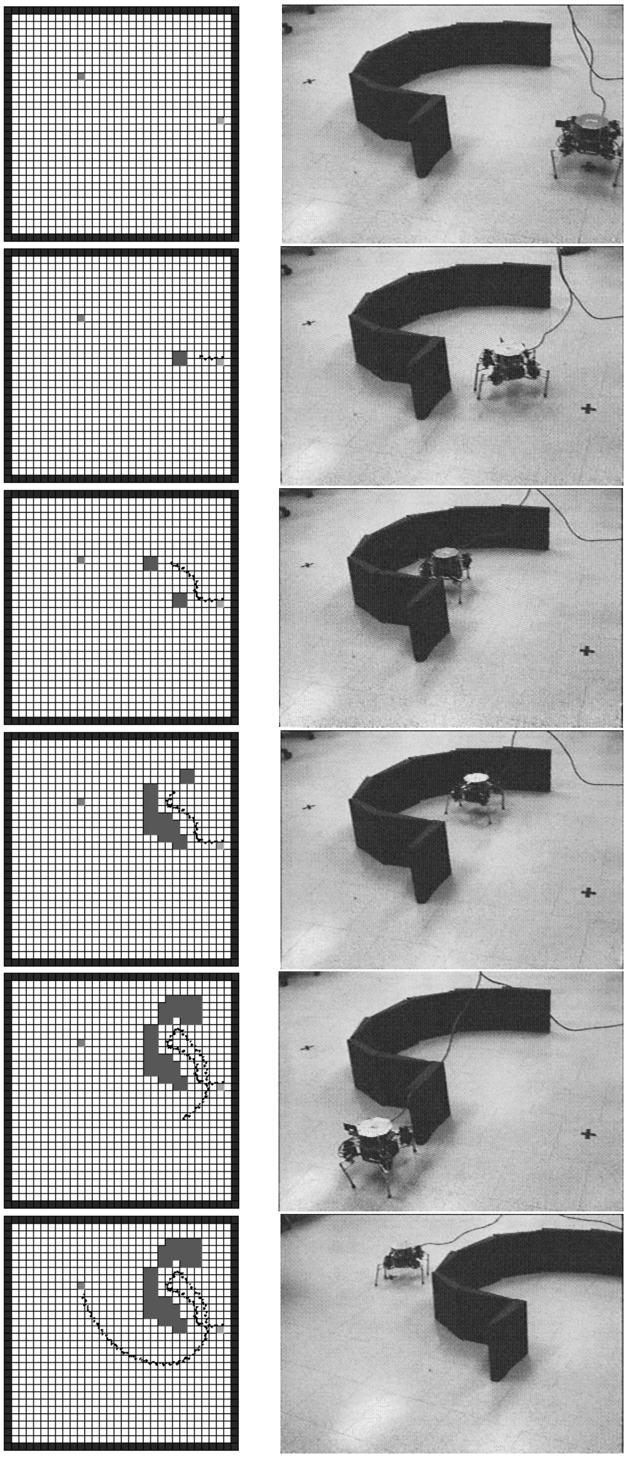

Figure 7: Walking Task in an Unknown Environment. Start and goal on the internal map (left) and actual robot (right) are indicated by initially shaded grid cells and crosses on the floor, respectively. The path of the robot is marked by black dots corresponding to the location of the center of mass.

5 Conclusions and Future Work

Most of the walking literature focuses on off-line gait planning, fixed periodic gaits, or traditional behavior-based control as a means of accomplishing walking tasks. All of these schemes require either long planning times for each step, large numbers of fixed geometric gaits, or complex organizations of carefully tailored task specific behavior rules in order to respond to changing walking objectives. In addition, most non-behavior based schemes do not take into account uncertainties in the execution or modeling of the terrain, and none of these approaches generalize easily to other robot kinematics.

The approach presented here avoids these limitations by constructing behavior on-line from a set of reactive and task-independent controllers, resulting in flexible and robust walking performance. Using two control gaits for rotating in place and walking with forward progress, the system is able to perform the task of walking from one point to another in an unknown environment while modeling obstacles and reacting to their presence. The underlying controllers are reusable and device independent and can thus be used to control not only quasistatic walking on robot platforms with arbitrary numbers of legs, but also to perform grasping and manipulation tasks on hand/arm platforms [6]. Moreover, not only individual basis controllers but also complete control gaits are largely platform independent. The control policies presented here could thus be transferred directly to other four-legged robot geometries.

To further investigate the potential of this control approach in the walking domain, the robot will be equipped with additional sensors including inclinometers, and torque and tactile sensors on the legs to provide the environmental information necessary for the traversal of irregular terrain. In addition, the possibilities of automatic generation of the required control gaits will be investigated.

References

[1] R. A. Brooks. A robot that walks; emergent behaviors from a carefully evolved network. *Neural Computation*, 1(2):355–363, 1989.

[2] J. A. Coelho Jr.. and R. A. Grupen. Effective multifingered grasp synthesis. Technical Report 94-12, CMPSCI Dept., University of Massachusetts, Amherst, February 1994.

[3] C. I. Connolly. Harmonic functions and collision probabilities. In *Proc. Int. Conf. Robotics Automat.*, San Diego, CA, May 1994. IEEE.

[4] C. I. Connolly and R. A. Grupen. The applications of harmonic functions to robotics. *J. Robotic Sys.*, 10(7):931–946, October 1993.

[5] P. de Santos and M. Jimenez. Generation of discontinuous gaits for quadruped walking vehicles. *J. Robotic Sys.*, 12(9):599–611, 1995.

[6] R. A. Grupen, M. Huber, J. A. Coelho Jr., and K. Souccar. Distributed control representation for manipulation tasks. *IEEE Expert*, 10(2):9–14, April 1995.

[7] S. Hirose. A study of design and control of a quadruped walking vehicle. *Int. J. Robotics Res.*, 3(2):113–133, 1984.

[8] K. Jeong, T. Yang, and J. Oh. A study on the support pattern of a quadruped walking robot for aperiodic motion. In *Proc. IROS*, pages 308–313, Pittsburgh, PA, August 1995. IEEE.

[9] W. MacDonald. Design and implementation of a four-legged walking robot. Senior honors thesis, ECE Dept., University of Massachusetts, Amherst, 1994.

[10] D. Pack and H. Kang. An omnidirectional gait control using a graph search method for a quadruped walking robot. In *Proc. Int. Conf. Robotics Automat.*, volume 1, pages 988–993, Nagoya, Japan, May 1995. IEEE.

[11] M. Raibert. *Legged Robots that Balance.* MIT Press, Cambridge, MA, 1986.

[12] P. J. Ramadge and W. M. Wonham. The control of discrete event systems. *Proceedings of the IEEE*, 77(1):81–97, January 1989.

[13] S.-M. Song and K. J. Waldron. *Machines That Walk: The Adaptive Suspension Vehicle.* MIT Press, Cambridge, MA, 1989.

[14] D. Wettergreen, H. Pangels, and J. Bares. Behavior-based gait execution for the Dante II walking robot. In *Proc. IROS*, pages 274–279, Pittsburgh, PA, August 1995. IEEE.

[15] T. Yoshikawa. *Foundations of Robotics : Analysis and Control.* MIT Press, Cambridge, MA, 1990.