Proceedings of the IEEE International Conference on Robotics and Automation 1997 ©IEEE

Force Control Command Synthesis for Assembly using a Discrete Event Framework

David Austin

david@faceng.anu.edu.au

Brenan McCarragher

brenan@faceng.anu.edu.au

Department of Engineering
Faculty of Engineering and Information Technology
The Australian National University
Canberra, Australia
Fax: int + 61 6 249 0506

Abstract

A new discrete event controller synthesis methodology for the successful convergence of assembly tasks is presented. The modelling of an assembly process as a hybrid dynamic system has been shown to be a very effective strategy to incorporate both the continuous and discrete natures of the interaction between the workpiece and its environment. Prior works have presented controller synthesis methodologies for velocity-controlled systems [8] and here we follow a similar structure for force-controlled systems. Force control is a natural paradigm for assembly as it is fundamentally compliant. Compliance reduces sensitivity to positioning errors, which are the most common source of failure in assembly. A non-trivial example is given which demonstrates the effectiveness of this method.

1 Introduction

Position errors have been shown to be the primary cause of failure in robotic assembly [9]. Instead, force control is a natural paradigm for assembly tasks as it is fundamentally compliant, virtually eliminating any dependence on position. However, work in the field of force control tends to concentrate on low-level details of the assembly problem, and as such have had limited success with significant effort. In contrast, we propose that work in assembly proceed at a more abstract level. In particular, this paper proposes a new methodology for discrete event controller synthesis for assembly tasks in which the continuous-time system is force controlled. The proposed controller synthesis methodology combines the advantages of discrete event control with the considerable body of work in the area of force control.

Automated assembly is a natural application of robotics and has been studied for many years. Whitney [13] developed quasi-static conditions to avoid wedging and jamming in the cylindrical *peg-in-hole* situation. O'Connor et al [9] studied the identification and classification of errors in automated assembly processes. Assembly is a special type of constrained motion system and a great deal of research has been conducted in the field of constrained motion systems. Hogan [4] developed impedance control for the manipulation of objects constrained by the environment. Mason [6] and Raibert and Craig [11] developed controllers which use both position and force control for the manipulation of constrained objects. The concept of using position and force control simultaneously has been extended by a number of researchers (e.g. [5] [12]) in search of a better control scheme for constrained motion systems. Unfortunately, assembly is still one of the most error-prone of robotic applications

Hybrid dynamic modelling is an ideal tool for assembly as it provides a good framework for modelling and analysis of abstract concepts linked to continuous systems. A hybrid dynamic system consists of a discrete event system interacting with a continuous time system. Usually, the discrete event system is a decision-maker or controller and operates at an abstract level. A simple example is a furnace system in a typical home. The thermostat is an abstract, discrete event system with two states "too cold" and "warm enough", whereas the house and furnace are continuous time systems. The control algorithm is to turn the furnace on when in the "too cold" state and turn the furnace off otherwise. Even this simple example illus-

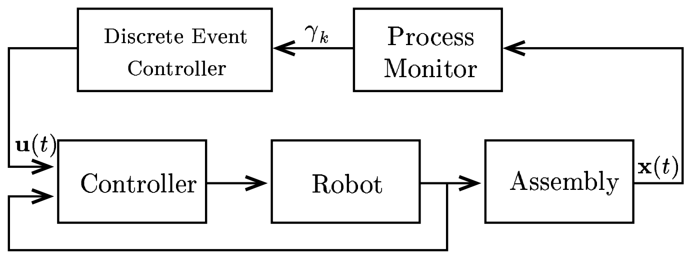

Figure 1: Block Diagram Representation of System

trates the important feature of hybrid dynamic models, which is to combine abstract concepts with continuous systems. There have been many wide-ranging applications of hybrid dynamic modelling, most notably in manufacturing systems and network protocols. To date, they have been used in detailed assembly tasks [8], in higher-level flexible manufacturing systems [3], and telephone network protocols [10].

Section 2 presents a framework for hybrid dynamic modelling of assembly tasks and introduces the discrete event controller. Section 3 presents the force control command synthesis methodology, including the development of constraints upon the control command and the solution of the constraints. Section 4 gives a three-dimensional force controller synthesis example involving a multi-point contact situation.

2 Hybrid Dynamic Modelling of Assembly Tasks

We consider a system which involves the motion of a rigid, polyhedral *workpiece* with possible constraints introduced by contact between the workpiece and a fixed, rigid and polyhedral *environment*. Systems of this type are typical of assembly processes. In a previous work, McCarragher [7] demonstrated that all possible states of contact between two polyhedral parts can be described as combinations of two basic contact types, namely edge-edge and surface-vertex contact. For example, a surface-surface contact may be described as the union of three surface-vertex contacts because three non-colinear points define a plane. Similarly, a edge-vertex contact may be described as the intersection of two edge-edge contacts because the only point where both edge-edge contacts can be valid is the vertex. Also important is the ability to represent distributed or muilti-point contacts as combinations of edge-edge and surface-vertex contacts. Hybrid dynamic modelling is particularly appropriate for assembly processes as there are a small number of possible combinations of edge-edge and surface-vertex contacts, and hence, we have dramatically reduced the complexity of the model by abstracting to a higher level.

We consider a specific type of hybrid dynamic system, consisting of a discrete event controller interfaced to a constrained motion system involving the motion of two polyhedral parts (as discussed above). The structure of the adopted hybrid dynamic model is as shown in Figure 1. The system consists of five parts: a continuous-time controller, a robot, an assembly, a process monitor and a discrete event controller. For this paper, the continuous-time controller is a force control loop, providing the ability to issue force/torque commands to the robot. The controller, robot and assembly are described by the differential equation

$$\dot{\mathbf{x}}(t) = \mathbf{f}\left(\mathbf{x}(t), \mathbf{u}(t)\right) \tag{1}$$

where $\mathbf{x}(t)$ is the *state* of the system and includes information such as the positions and velocities of the workpiece and the contact forces with the environment. $\mathbf{u}(t)$ is the input to the system or the *control command.* The process monitor uses the continuous signal $\mathbf{x}(t)$ to detect contact changes in the system and determine corresponding discrete events. From the discrete events, the discrete state γ of the system is determined. For our purposes, the discrete states of the system correspond to the possible combinations of edge-edge and surface-vertex contacts. The process monitor is defined as the map ψ:

$$\gamma_k = \psi\left(\mathbf{x}(t)\right) \tag{2}$$

where γ_k is the k^{th} discrete state of the system. The discrete event controller is the part of the system which determines the appropriate command to issue, based upon the discrete state of the system. The discrete event controller is described by the map ϕ:

$$\mathbf{u}(t) = \phi\left(\gamma_k\right) \tag{3}$$

For the purposes of this paper, $\mathbf{u}(t)$ will be a force/torque vector or a *wrench* [1]

$$\mathbf{u}(t) = \begin{bmatrix} \mathbf{F} \\ \tau \end{bmatrix} \qquad t_k \leq t < t_{k+1} \tag{4}$$

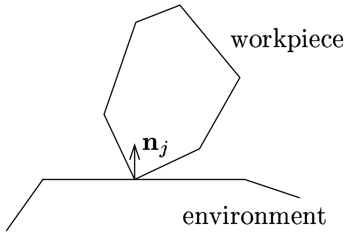

Figure 2: Contact Normal Definition

where $\mathbf{F}$ is the force and τ the torque command and t_k and t_{k+1} are the times of the k^{th} and $k+1^{th}$ events, respectively.

As the workpiece interacts with the environment, the free-space dynamics of (1) become constrained. The geometric constraints may be expressed as

$$\mathbf{g}_j\left(\mathbf{y}(t)\right) = 0 \tag{5}$$

where $\mathbf{g}_j$ is the constraint function for the j^{th} edge-edge or surface-vertex contact and $\mathbf{y}(t)$ is the position of the workpiece. Equivalently, we may express the contact situation as a statement that the perpendicular contact force is greater than zero for some contact pair.

$$\mathbf{F}_j.\mathbf{n}_j > \mathbf{0} \tag{6}$$

where $\mathbf{F}_j$ is the contact force applied by the environment to the workpiece at the j^{th} edge-edge or surface-vertex contact. $\mathbf{n}_j$ is the unit normal associated with the j^{th} edge-edge or surface-vertex pair. For an edge-edge contact the normal is defined to be a vector perpendicular to both edges and for a surface-vertex contact the normal is defined as the normal to the surface. The direction of the normal is chosen such that it points towards the workpiece and away from the environment, as shown in Figure 2.

3 Control Command Synthesis

The *control commands* are the input to the continuous-time plant from the discrete event controller. For our purposes, the control commands depend solely upon the discrete state of the assembly process. Under the discrete event framework, the control command changes when events occur, and as a result, a control command can only achieve a single contact change. The control commands are determined by first establishing a *desired event* for each discrete state. The desired event is chosen such that the system moves towards the completion of the assembly. The series of desired events gives an event trajectory which accomplishes the assembly task. For any given discrete state, we use the desired event and geometric considerations of the workpiece and environment to establish conditions on the command to be executed.

There are three conditions upon which the control command (3) is selected. First, the *maintaining condition* ensures that the currently active constraints remain satisfied, if desired. Secondly, we require a condition that ensures that we move towards the completed assembly. In a discrete event framework we cannot force an event to occur, instead we must enable it and wait for the next event. The second condition is the *enabling condition* and is a necessary condition that enables the next desired discrete event γ_{k+1}. Finally, we wish to prevent undesired events which would cause the system to move away from the completion of the assembly. The *disabling condition* is a sufficient condition that ensures an undesired discrete event γ_l is not allowed to occur.

In prior works [8], the three conditions have been developed as constraints on the velocity of the workpiece and a velocity command has been determined from the constraints. In this paper, we consider a force-controlled system for which we must determine force commands, and hence, the three conditions will be constraints on the force command.

3.1 Maintaining Condition

The first possible task of the controller is to ensure that the control commands satisfy a currently active geometric constraint or contact. To maintain a contact, the normal component of the contact force must be positive:

$$\mathbf{F}_j \,.\, \mathbf{n}_j > 0 \tag{7}$$

where $\mathbf{F}_j$ is the contact force for the j^{th} edge-edge or surface-vertex contact and $\mathbf{n}_j$ is the unit normal, as defined in Section 2. To derive admissible force commands that satisfy the geometric constraint, we analyse the forces applied to the workpiece. For a single contact situation, $\mathbf{F}_j$ depends solely upon the applied force and torque.

Consider the forces on the workpiece as presented in Figure 3. The workpiece is a simple system with a wrench applied at point O and a contact force resulting from the interaction with the environment. Here we include frictional forces, as indicated by the tangential component of $\mathbf{F}_j$ in Figure 3. The equations of motion of the workpiece may be written as

$$\mathbf{F} + \mathbf{F}_j = \mathbf{Ma} \tag{8}$$

$$\tau + \mathbf{r}_j \times \mathbf{F}_j = \mathbf{I}_o \alpha \tag{9}$$

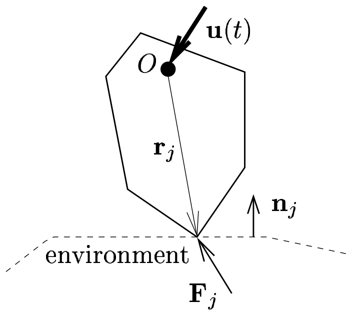

Figure 3: Forces Applied to the Workpiece in a Single-Contact Situation

where $\mathbf{M}$ is the mass matrix for the workpiece, $\mathbf{a}$ is the acceleration vector, $\mathbf{I}_o$ is the inertia matrix about point O (see Figure 3) and α is the angular acceleration of the workpiece about point O. In most cases the mass of the workpiece and the accelerations involved will be small enough that $\mathbf{Ma} \approx 0$ and $\mathbf{I}_o\alpha \approx 0$. However, the effects of the mass and moment of inertia may be cancelled by writing

$$\mathbf{F}' = \mathbf{Ma} - \mathbf{F} \quad (10)$$
$$\tau' = \mathbf{I}_o\alpha - \tau \quad (11)$$

and finding conditions on $\mathbf{F}'$ and τ'. We substitute into (8) and (9) to give

$$\mathbf{F}' = \mathbf{F}_j \quad (12)$$
$$\tau' = \mathbf{r}_j \times \mathbf{F}_j \quad (13)$$

To determine conditions upon the control command we find the resultant contact force for an arbitrary applied wrench and then combine with the requirement of (7). We determine the minimum wrench resulting in a unit contact force and then we can determine the contact force resulting from an arbitrary wrench. We find the minimum wrench by setting $\mathbf{F}_j = \mathbf{n}_j$ and using the equations of motion (12) and (13) to find the components of the wrench.

$$\hat{\mathbf{F}} = \mathbf{n}_j \quad (14)$$
$$\hat{\tau} = \mathbf{r}_j \times \mathbf{n}_j \quad (15)$$

where $\hat{\mathbf{F}}$ is a force which results in a unit perpendicular contact force and $\hat{\tau}$ is the minimum torque vector which results in a unit perpendicular contact force. Thus we determine the perpendicular contact force for an arbitrary applied force $\mathbf{F}'$ by determining the number of multiples of $\hat{\mathbf{F}}$ in $\mathbf{F}'$. Mathematically,

$$\mathbf{F}_j \,.\, \mathbf{n}_j = \frac{\mathbf{F}' \,.\, \hat{\mathbf{F}}}{\left|\hat{\mathbf{F}}\right|^2} = \mathbf{F}' \,.\, \mathbf{n}_j \quad (16)$$

Equation (16) can also be determined directly from (12). Using a similar method, we may determine the perpendicular contact force for an arbitrary applied torque τ'. The perpendicular contact force for an arbitrary applied force $\mathbf{F}'$ is given by the number of multiples of $\hat{\mathbf{F}}$ in $\mathbf{F}'$. i.e.

$$\mathbf{F}_j \,.\, \mathbf{n}_j = \frac{\tau' \,.\, \hat{\tau}}{|\hat{\tau}|^2} \quad (17)$$

In order to satisfy the maintaining condition (7) we require that the net contact force resulting from the applied wrench be positive, which may be written as a condition on the wrench:

$$\begin{bmatrix} \mathbf{n}_j & \frac{\hat{\tau}_j}{|\hat{\tau}_j|^2} \end{bmatrix} \begin{bmatrix} \mathbf{F}' \\ \tau' \end{bmatrix} > 0 \quad (18)$$

Equation (18) is our maintaining condition in that it must be satisfied to maintain the contact or geometric constraint. It is basically a requirement on the applied force and torque so that the contact force remains positive.

3.2 Enabling Condition

In addition to determining motion that maintains a constraint, it is desired to determine the motion such that the workpiece encounters the next discrete state γ_{k+1}. To enable a "gain of contact" event, we must cause the contact force to become positive by issuing a force command satisfying

$$\begin{bmatrix} \mathbf{n}_j & \frac{\hat{\tau}_j}{|\hat{\tau}_j|^2} \end{bmatrix} \begin{bmatrix} \mathbf{F}' \\ \tau' \end{bmatrix} > 0 \quad (19)$$

and the force controller will endeavour to drive the system so that contact is gained.

To enable a "loss of contact" event, we must cause the contact force to become zero by issuing a force command satisfying

$$\begin{bmatrix} \mathbf{n}_j & \frac{\hat{\tau}_j}{|\hat{\tau}_j|^2} \end{bmatrix} \begin{bmatrix} \mathbf{F}' \\ \tau' \end{bmatrix} = 0 \quad (20)$$

and the force controller will drive the system so that contact is lost.

3.3 Disabling Condition

The third condition, the disabling condition, is used to prevent unwanted gains of contact and is derived directly from the enabling condition. Since (19) is a necessary condition for a gain of contact to occur, a sufficient condition to prevent a gain of contact is obtained by requiring the contact force to remain zero

$$\begin{bmatrix} \mathbf{n}_j & \frac{\hat{\tau}_j}{|\hat{\tau}_j|^2} \end{bmatrix} \begin{bmatrix} \mathbf{F}' \\ \tau' \end{bmatrix} = 0 \quad (21)$$

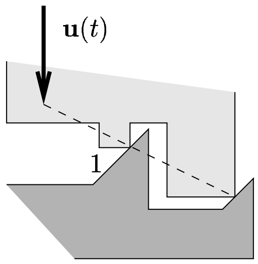

Figure 4: Example of Lack of "Spread" in a Multi-Contact Situation

3.4 Solving for the Control Command

The desired event determines which of the above conditions should be applied for each possible edge-edge or surface-vertex contact. The maintaining condition is (18) and is used when it is desired to maintain a contact. Note, when it is desired to immediately violate the current constraint by breaking the contact, the maintaining condition is not used. The enabling condition is (19) or (20) and is used to enable the gain or loss of a contact. The disabling condition is (21) and is used to prevent unwanted gains of contact. From the desired event, we now find a set of conditions on the control command, one for each possible edge-edge or surface-vertex contact. The control command is then determined by satisfying this set of conditions. Any method for satisfying the set of constraints will yield an acceptable force command. One method [2], which uses a search technique to maximise the minimum distance to each constraint for maximum robustness, is suggested.

3.5 Multi-Contact Situations

Analysis becomes much more complex when multiple contacts between the workpiece and its environment are active. Using the work of McCarragher [7] we are able to decompose multi-point contacts into combinations of edge-edge and surface-vertex contact. However, the analysis beyond this becomes intractable. Whitney [13] considers two point contact only in two dimensions.

Intuitively we would argue that the conditions derived above (equations (18) (19) (20) and (21)) can be used for each contact of a multi-contact situation provided

1. frictional forces are small compared to perpendicular contact forces, and
2. the vectors $\begin{bmatrix} \mathbf{n}_j & \frac{\hat{r}_j}{|\hat{r}_j|^2} \end{bmatrix}$ are linearly independent.

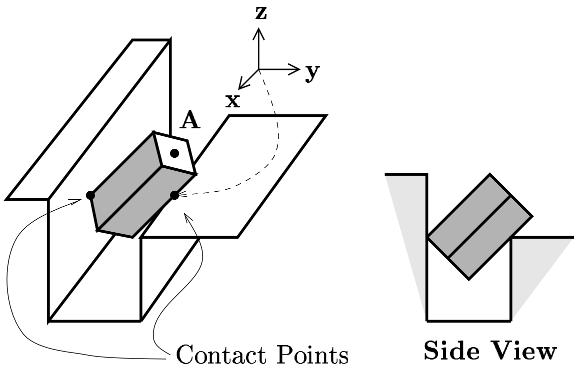

Figure 5: Example: 3D Peg and Trough in Dual Contact Configuration

The first condition states the requirement that the only significant contact forces lie along the direction of the contact normals (friction accounts for the tangential component, if any). This requirement ensures that the forces resulting from the applied wrench will dominate.

The second condition is a statement of the requirement that the contact points must be "spread". Figure 4 demonstrates an example with poor spread. For the system shown in Figure 4 the vectors $\begin{bmatrix} \mathbf{n}_1 & \frac{\hat{r}_1}{|\hat{r}_1|^2} \end{bmatrix}$ and $\begin{bmatrix} \mathbf{n}_2 & \frac{\hat{r}_2}{|\hat{r}_2|^2} \end{bmatrix}$ are equal. In this case, it would not be possible to solve for a control command which maintained contact 1 and lost contact 2 because (18) and (20) would be conflicting conditions. The conflict is possible only when the contact normals are linearly dependent and the $\mathbf{r}_j$ vectors are linearly dependent. As such, Figure 4 represents degenerate or singular cases which only occur at a single point and can be avoided with proper discrete event planning.

4 Example

Consider the peg and trough as shown in Figure 5. The peg is a rectangular parallel-piped with a length of $4.24cm$ and a square cross-section with side $2cm$. The force and torque commands are applied to the peg at point A. Let the trough have a width of $4cm$ and let the peg be engaged with the trough rotated about its long axis 45^o and about the axis of the trough 45^o. In this orientation, there are two points of contact as shown in Figure 5. It is desired to maintain both points of contact.

Considering the surface-vertex contact first, we see that the contact normal $\mathbf{n}_1 = \begin{bmatrix} 0 & 1 & 0 \end{bmatrix}$, and the radius vector from $\mathbf{A}$ to the point of contact is $\mathbf{r}_1 = \begin{bmatrix} 0 & -0.02 & -0.02 \end{bmatrix}$. From equation (15), $\hat{\tau}_1 = \begin{bmatrix} -0.02 & 0 & 0 \end{bmatrix}$. We wish to maintain the contact so equation (18) is used. The maintaining condition for the surface-vertex contact is then

$$\begin{bmatrix} 0 & 1 & 0 & -50 & 0 & 0 \end{bmatrix} \begin{bmatrix} \mathbf{F}' \\ \tau' \end{bmatrix} > 0 \qquad (22)$$

Now considering the edge-edge contact, we see that $\mathbf{n}_2 = \begin{bmatrix} 0 & -\frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \end{bmatrix}$, $\mathbf{r}_2 = \begin{bmatrix} 0 & 0 & -0.02 \end{bmatrix}$. Hence, $\hat{\tau}_2 = \begin{bmatrix} 0.01\sqrt{2} & 0 & 0 \end{bmatrix}$. We wish to maintain the contact so equation (18) is used. The maintaining condition for the edge-edge contact is then

$$\begin{bmatrix} 0 & 0.707 & 0.707 & 70.7 & 0 & 0 \end{bmatrix} \begin{bmatrix} \mathbf{F}' \\ \tau' \end{bmatrix} > 0 \qquad (23)$$

We solve (22) and (23) using a search technique to maximise the minimum distance to each constraint. Note, we also impose a unit magnitude constraint to ensure that the wrench remains bounded. The resulting wrench may then be scaled as desired. The resulting wrench is

$$\begin{bmatrix} \mathbf{F}' \\ \tau' \end{bmatrix} = \begin{bmatrix} 0.15 & 0.41 & 0.80 & 0.00 & 0.25 & 0.34 \end{bmatrix} \qquad (24)$$

If we assume that $\mathbf{a} = 0$ and $\alpha = 0$ then, using equations (10) and (11), we have

$$\begin{bmatrix} \mathbf{F} \\ \tau \end{bmatrix} = \begin{bmatrix} -0.15 & -0.41 & -0.80 & 0 & -0.25 & -0.34 \end{bmatrix} \qquad (25)$$

and clearly, the three components of interest satisfy the requirements to maintain contact. The remaining components are non-zero due to numerical errors within the search algorithm.

Even this simple example demonstrates the effectiveness of the controller synthesis technique in determining suitable force commands to achieve desired discrete state transitions. Despite the lack of formal proof for multi-contact situations, this example illustrates that the controller synthesis technique is valid for practical multi-contact situations.

5 Conclusions

In this paper, we have presented a new discrete event controller synthesis methodology for hybrid dynamic systems involved in constrained motion tasks. This synthesis methodology provides for the generation of discrete event controllers for force-controlled systems whereas, prior works have only provided for velocity-controlled systems. A complex practical example has demonstrated the effectiveness of the synthesis methodology.

The possibility of combining the velocity and force command synthesis methodologies to provide a new synthesis methodology for a system which is both position and force controlled is under investigation.

References

[1] R. S. Ball. *The Theory of Screws.* Cambridge University Press, 1900.

[2] M. J. Best and K. Ritter. *Linear Programming: Active Set Analysis and Computer Programs.* Prentice-Hall, Inc., 1985.

[3] S. Gershwin. Hierarchical flow control: A framework for scheduling and planning discrete events in manufacturing systems. In *Proceedings IEEE, Special Issue on Dynamics of Discrete Event Systems*, volume 77-1, pages 195–209, January 1989.

[4] N. Hogan. Impedance control: An approach to manipulation, parts i, ii and iii. *Journal of Dynamic Systems, Measurement and Control*, 107:1–24, March 1985.

[5] D. Jeon and M. Tomizuka. Learning hybrid force and position control of robot manipulators. *IEEE Transactions on Robotics and Automation*, 9:423–431, 1996.

[6] M. T. Mason. Compliance and force control for computer-controlled manipulators. *IEEE Transactions on Systems, Man, and Cybernetics*, 11(6):418–432, June 1981.

[7] B. McCarragher. Task primitives for the discrete event modeling and control of 6-dof assembly tasks. *IEEE Transactions on Robotics and Automation*, 12(2):280–289, April 1996.

[8] B. J. McCarragher and H. Asada. The discrete event modelling and trajectory planning of robotic assembly tasks. *ASME Journal of Dynamic Systems, Measurement, and Control*, 117(3):394–400, September 1995.

[9] R. F. O'Connor, C. Baber, M. Musri, and H. Ekerol. Identification, classification and management of errors in automated assembly tasks. *Intl. Journal of Production Research*, 31:1853–1863, 1991.

[10] A. Overkamp. Design and verification of a communication protocol for a telephone network. In *Belgian-French-Netherlands Summer School on Discrete Event Systems*, June 1993.

[11] M.H. Raibert and J.J. Craig. Hybrid position/force control of manipulators. *Journal of Dynamic Systems, Measurement, and Control*, 103:126–133, 1981.

[12] B. Siciliano and L. Villani. A force/position regulator for robot manipulators without velocity measurements. In *Proc. 1996 IEEE Intl. Conf. on Robotics and Automation*, pages 2567–2572, 1996.

[13] D. E. Whitney. Quasi-static assembly of compliantly supported rigid parts. *ASME Journal of Dynamic Systems, Measurement and Control*, 1982.