ARM SIGNATURE IDENTIFICATION

Henry W. Stone, Arthur C. Sanderson, and Charles P. Neuman

Department of Electrical and Computer Engineering and Robotics Institute
Carnegie-Mellon University
Pittsburgh, PA 15213

The positioning accuracy of commercially-available industrial robotic manipulators depends upon a kinematic model which describes the robot geometry in a parametric form. Manufacturing errors in machining and assembly of manipulators lead to discrepancies between the design parameters and the physical structure. Improving the kinematic performance thus requires identification of the *actual* kinematic parameters of each individual robot. This identification of the individual kinematic parameters is called the *arm signature* which is then incorporated into the manipulator's controller to improve positional accuracy. In this paper, an approach, based on a new parametric model of the kinematics, is introduced for arm signature identification. The *S-Model* utilizes $6 \cdot n$ parameters to describe the robot geometry and offers advantages for identification by decomposing the parameters into individually identified subsets. The S-Model parameters are then mapped into the equivalent *Denavit-Hartenberg* parameters for implementation into the controller. The S-Model arm signature identification algorithm can be implemented with relatively simple sensors and improves accuracy through statistical averaging. This algorithm has been implemented with an external ultrasonic range sensor to measure robot end-effector positions. Experimental results of arm signature identification of seven Unimation/Westinghouse Puma 560 robots demonstrated an average reduction in positioning error by a factor of 5-10 for a spectrum of representative test tasks.

1 Introduction

Commercial robotic manipulators currently utilize *feedforward* control which depends upon a parametric kinematic robot model. Since manufacturing errors are random, each manipulator possesses unique kinematics and the positioning performance of these robots degrades significantly when there is a mismatch between the actual robot and the model which implements the kinematic control. This mismatch is especially prevalent in manipulators with revolute joints in which small manufacturing errors produce significant errors between the actual and predicted positions and orientations of the end-effector. Practical methods for identifying the actual kinematic models of n degree-of-freedom robotic manipulators are required to improve their kinematic performance. We refer to these models, whose parameters are estimates of the unique kinematic parameters, as *arm signatures*. Once identified, the arm signatures can be used to synthesize and implement control algorithms to improve kinematic performance.

Kinematic parameter identification algorithms have been proposed in the literature [1, 4, 5, 9]. Our S-Model identification algorithm identifies the parameters of a nonlinear kinematic model. We apply sequential linear least-squares algorithms to identify the parameters of manipulator kinematic features (two features for each revolute joint and one feature for each prismatic joint). These features contain the essential information to model completely the kinematics of a manipulator. We identify each feature in an independent procedure and then extract the Denavit-Hartenberg parameters. To estimate the features, we measure the three dimensional cartesian positions of target points mounted on the manipulator links. The measurements are collected and processed automatically by the robot controller under software control. The effects of measurement noise are reduced through statistical averaging embedded within the estimation algorithms. Since the sensor system is independent of the physical manipulator, we eliminate the need for elaborate and costly fixturing or manual positioning of the end-effector. We have implemented and applied our S-Model identification algorithm to seven Unimation/Westinghouse Puma 560 robots. Evaluation of the experimental results has demonstrated consistent improvements in their kinematic performance.

This paper is organized as follows. In Section 2, we introduce the S-Model and in Section 3, we describe our identification algorithm. Then, in Section 4, we highlight our hardware implementation and illustrate the improved kinematic performance of the seven Puma 560 robots. Concluding remarks appear in Section 5.

2 Formulation of the S Model

2.1 Notation

We use standard robotic notation. Uppercase boldface letters (e.g., $\mathbf{T}_i$ and $\mathbf{S}_i$) denote (4x4) homogeneous transformation matrices. These matrices describe the relative position and orientation of two cartesian coordinate systems. Uppercase script letters (e.g., $\mathcal{T}_i$, and $\mathcal{S}_i$) denote the symbolic name of a cartesian coordinate system. Lowercase letters (e.g., $\mathbf{d}_i$ and β_i) denote scalar parameters.

2.2 The Denavit-Hartenberg Model

In the robotics literature, the (4x4) homogeneous transformation matrix

$$\mathbf{T}_n = \mathbf{A}_1 \cdot \mathbf{A}_2 \cdot \ldots \cdot \mathbf{A}_n \tag{1}$$

defines the position and orientation of a coordinate frame fixed relative to the last link (n^{th} link) of a manipulator with respect to a coordinate frame fixed relative to the base of the manipulator. The Denavit-Hartenberg link coordinate frames, $\mathcal{T}_i$ for $i=1,\ldots,n$, are specified so that the forward transformation matrices $\mathbf{A}_i$ are prescribed by

$$\mathbf{A}_i = Rot(z,\theta_i)Trans(0,0,\mathbf{d}_i)Trans(\mathbf{a}_i,0,0)\cdot Rot(x,\alpha_i) \quad . \tag{2}$$

In (2), $\mathbf{A}_i$, which is the transformation from coordinate frame $\mathcal{T}_{i-1}$ to coordinate frame $\mathcal{T}_i$, is a function of the *four* Denavit-Hartenberg parameters, θ_i , $\mathbf{d}_i$, $\mathbf{a}_i$, and α_i . An n degree-of-freedom manipulator requires the specification of $4 \cdot n$ parameters. The parameters $\mathbf{a}_i$ and α_i are the link length and twist, respectively [2]. The parameters θ_i and $\mathbf{d}_i$ are offsets or joint positions depending upon whether the i^{th} joint is revolute or prismatic. In Figure 1, we illustrate the geometrical interpretation of the Denavit-Hartenberg parameters θ_i , $\mathbf{d}_i$, $\mathbf{a}_i$, and α_i for a *revolute* joint. For the companion illustration of the Denavit-Hartenberg parameters for a *prismatic* joint, we refer the reader to Paul [6].

The characteristics of the Denavit-Hartenberg model are immediate consequences of the Denavit-Hartenberg convention applied to specify the link coordinate frames. The Denavit-Hartenberg convention follows from a geometrical analysis of the spatial relationships between consecutive joint axes. Knowledge of the position and orientation of the joint axes is thus fundamental to the kinematic modeling of manipulators. For the development of our S-Model (in Section 2.3), we review the link coordinate frame assignments according to the Denavit-Hartenberg convention:

- The Z axis of coordinate frame $\mathcal{T}_{i-1}$ must be parallel to the joint i axis;
- The origin of coordinate frame $\mathcal{T}_{i-1}$ must lie on the joint i axis at the intersection point of the common normal between the joint i-1 and joint i axes, and the joint i axis;
- The X axis of coordinate frame $\mathcal{T}_{i-1}$ must be parallel to the common normal between the joint i-1 and joint i axes. The positive direction of the X axis points towards the joint i axis;
- The Y axis of coordinate frame $\mathcal{T}_{i-1}$ is defined by the vector cross product of the Z axis unit direction vector with the X axis unit direction vector;
- If the joint i and joint $i+1$ axes intersect, the point of intersection is the origin of the $\mathcal{T}_{i-1}$ coordinate frame;
- If the joint i and joint $i+1$ axes are parallel the origin of the coordinate frame, $\mathcal{T}_{i-1}$ is chosen so that the joint distance $\mathbf{d}_{i+1}$ for the next link is equal to zero;
- The origin of the base link coordinate frame $\mathcal{T}_0$, coincides with the origin of the link 1 coordinate frame $\mathcal{T}_1$; and
- The origin of the last coordinate frame $\mathcal{T}_n$, coincides with the origin of the next to last coordinate frame $\mathcal{T}_{n-1}$.

These assignments guarantee the functional form of the Denavit-Hartenberg model in (1).

For a *revolute* joint, θ_i is the joint i position, and the three parameters, $\mathbf{d}_i$, $\mathbf{a}_i$, and α_i are constants. For a *prismatic* joint, $\mathbf{d}_i$

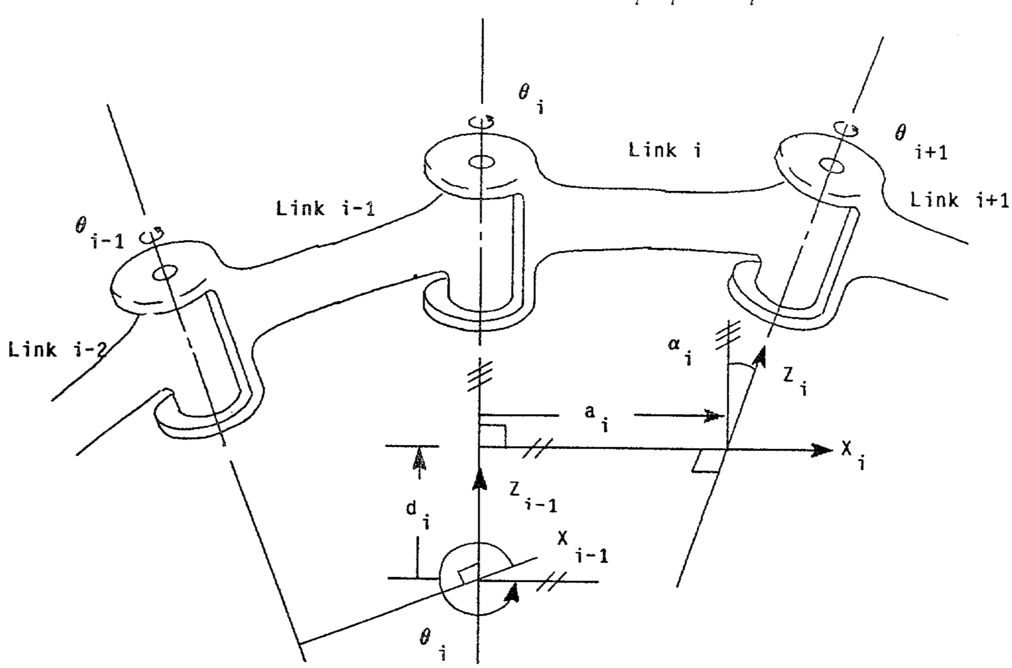

Figure 1: Denavit-Hartenberg Parameters for a Revolute Joint

is the joint i position, and the three parameters θ_i, $\mathbf{a}_i$, and α_i are constants. Manipulator joint encoders are calibrated to insure that the encoder outputs match the Denavit-Hartenberg joint positions (i.e., θ_i for a revolute joint and $\mathbf{d}_i$ for a prismatic joint). Without this calibration, constant offsets must be introduced to specify the difference between the joint positions measured by the encoder and the joint positions defined by the Denavit-Hartenberg model. When all of the joint positions are *zero*, we say that the manipulator is in the *Denavit-Hartenberg Zero Configuration*.

2.3 The S-Model

Like the Denavit-Hartenberg model, the S-Model is a completely general method for describing and characterizing kinematics of robotic manipulators. In the S-Model, the matrix

$$\mathbf{S}_n = \mathbf{B}_1 \cdot \mathbf{B}_2 \cdot \ldots \cdot \mathbf{B}_n \tag{3}$$

defines the position and orientation of a coordinate frame fixed relative to the last (n^{th}) link of a manipulator with respect to a coordinate frame fixed relative to the base link. The *general transformation matrices* $\mathbf{B}_i$ in (3) are (4x4) homogeneous transformation matrices. The $\mathbf{B}_i$ and $\mathbf{S}_n$ matrices in (3) are analogous to the $\mathbf{A}_i$ and $\mathbf{T}_n$ matrices of the Denavit-Hartenberg model in (1). The symbolic name $\mathcal{S}_i$ signifies the i^{th} link coordinate frame defined by the S-Model. The transformation matrix $\mathbf{B}_i$ describes the relative transformation between the $\mathcal{S}_{i-1}$ and $\mathcal{S}_i$ coordinate frames (measured with respect to the $\mathcal{S}_{i-1}$ coordinate frame). In the S-Model, *six* parameters β_i, $\bar{\mathbf{d}}_i$, $\bar{\mathbf{a}}_i$, $\bar{\alpha}_i$, γ_i, and $\mathbf{b}_i$ define the transformation matrix

$$\mathbf{B}_i = Rot(z,\beta_i)Trans(0,0,\bar{\mathbf{d}}_i)Trans(\bar{\mathbf{a}}_i,0,0) \tag{4}$$
$$Rot(x,\bar{\alpha}_i)Rot(z,\gamma_i)Trans(0,0,\mathbf{b}_i) \quad .$$

To specify the S-Model for an n degree-of-freedom manipulator thus requires $6 \cdot n$ parameters.

To insure that the kinematics of the manipulator can be modeled by (3), we introduce an S-Model convention to define the allowable locations of the link coordinate frames. Because each joint is specified by six parameters, the S-Model convention is less restrictive than the Denavit-Hartenberg convention.

The following four assignments (which are a subset of the Denavit-Hartenberg convention reviewed in Section 2.2) specify the locations of the S-Model link coordinate frames:

- The Z axis of the link coordinate frame $\mathcal{S}_{i-1}$ must be parallel to the joint i axis in the direction defined by the positive sense of the rotation or translation of the i^{th} joint;
- The origin of the coordinate frame $\mathcal{S}_{i-1}$ must lie on the joint i axis;
- The Z axis of the last oordinate frame $\mathcal{S}_n$ is parallel to the Z axis of the next to last coordinate frame $\mathcal{S}_{n-1}$; and
- The origin of the last coordinate frame $\mathcal{S}_n$ lies on the joint n-1 axis.

There are two fundamental distinctions between the Denavit-Hartenberg link coordinate frame $\mathcal{T}_i$ and the S-Model link coordinate frame $\mathcal{S}_i$. First, in contrast to the origin of $\mathcal{T}_i$, the location of the origin of $\mathcal{S}_i$ on the joint $i+1$ axis is arbitrary. Second, the direction of the X axis of $\mathcal{S}_i$ must only be orthogonal to the Z axis. The arbitrary location of the origin of $\mathcal{T}_i$ along the joint axis and the arbitrary orientation of the X axis of $\mathcal{S}_i$ provide an infinite number of link coordinate frames $\mathcal{S}_0$ through $\mathcal{S}_n$ which satisfy the S-Model convention.

The transformation matrix $\mathbf{B}_i$ can be formulated from the geometry of Figure 2. In Figure 2, we apply the S-Model convention to define a pair of link coordinate frames $\mathcal{S}_{i-1}$ and $\mathcal{S}_i$. For comparison, we also depict the Denavit-Hartenberg coordinate frames $\mathcal{T}_{i-1}$ and $\mathcal{T}_i$. The angle γ_i is defined as the *angular* displacement between the X axes of the Denavit-Hartenberg coordinate frame $\mathcal{T}_i$ and the S-Model coordinate frame $\mathcal{S}_i$. The parameter $\mathbf{b}_i$ is defined as the *linear* displacement between the origins of the Denavit-Hartenberg coordinate frame $\mathcal{T}_i$ and the S-Model link coordinate frame $\mathcal{S}_i$. (γ_i is positive if the displacement is in the direction of the Z axis of joint i.)

The transformation matrix $\mathbf{B}_i$ specifies the spatial transformation between the $\mathcal{S}_{i-1}$ and $\mathcal{S}_i$ link coordinate frames for both prismatic and revolute joints. From Figure 2 and the definitions of γ_i, $\mathbf{b}_i$, and $\mathbf{A}_i$, the transformation matrix $\mathbf{B}_i$ is the product

$$\mathbf{B}_i = Rot(z,-\gamma_{i-1})Trans(0,0,-\mathbf{b}_{i-1})[Rot(z,\theta_i)\cdot$$
$$Trans(0,0,\mathbf{d}_i)Trans(\mathbf{a}_i,0,0)Rot(x,\alpha_i)]\cdot$$
$$Rot(z,\gamma_i)Trans(0,0,\mathbf{b}_i) \quad . \tag{5}$$

The first transformation, $Rot(z,-\gamma_{i-1})$, aligns the orientation of the axes of coordinate frames $\mathcal{S}_{i-1}$ and $\mathcal{T}_{i-1}$. The second transformation, $Trans(0,0,-\mathbf{b}_i)$, translates the origin of $\mathcal{S}_{i-1}$ so that it coincides with the origin of the Denavit-Hartenberg coordinate frame $\mathcal{T}_{i-1}$. The four bracketed transformations in (5) define the Denavit-Hartenberg matrix $\mathbf{A}_i$. (The parameters θ_i, $\mathbf{d}_i$, $\mathbf{a}_i$, and α_i are the Denavit-Hartenberg parameters for link i.) These four matrices transform coordinate frame $\mathcal{T}_{i-1}$ to the Denavit-Hartenberg coordinate frame $\mathcal{T}_i$. In analogy with the first two transformations, the cascade $Rot(z,\gamma_i)Trans(0,0,\mathbf{b}_i)$ transforms the Denavit-Hartenberg coordinate frame $\mathcal{T}_i$ to the S-Model link coordinate frame $\mathcal{S}_i$.

Upon applying the properties of homogeneous transformations [6], $\mathbf{B}_i$ in (5) simplifies to

$$\mathbf{B}_i = Rot(z,\theta_i-\gamma_{i-1})Trans(0,0,\mathbf{d}_i-\mathbf{b}_{i-1})\cdot$$
$$Trans(\mathbf{a}_i,0,0)Rot(x,\alpha_i)Rot(z,-\gamma_i)\cdot$$
$$Trans(0,0,\mathbf{b}_i) \quad . \tag{6}$$

Since (4) and (5) are equivalent,

$$\beta_i = \theta_i - \gamma_{i-1} \tag{7}$$

$$\bar{\mathbf{d}}_i = \mathbf{d}_i - \mathbf{b}_{i-1} \tag{8}$$

where the joint rotational offset γ_i and the joint translational offset $\mathbf{b}_i$ are constant parameters.

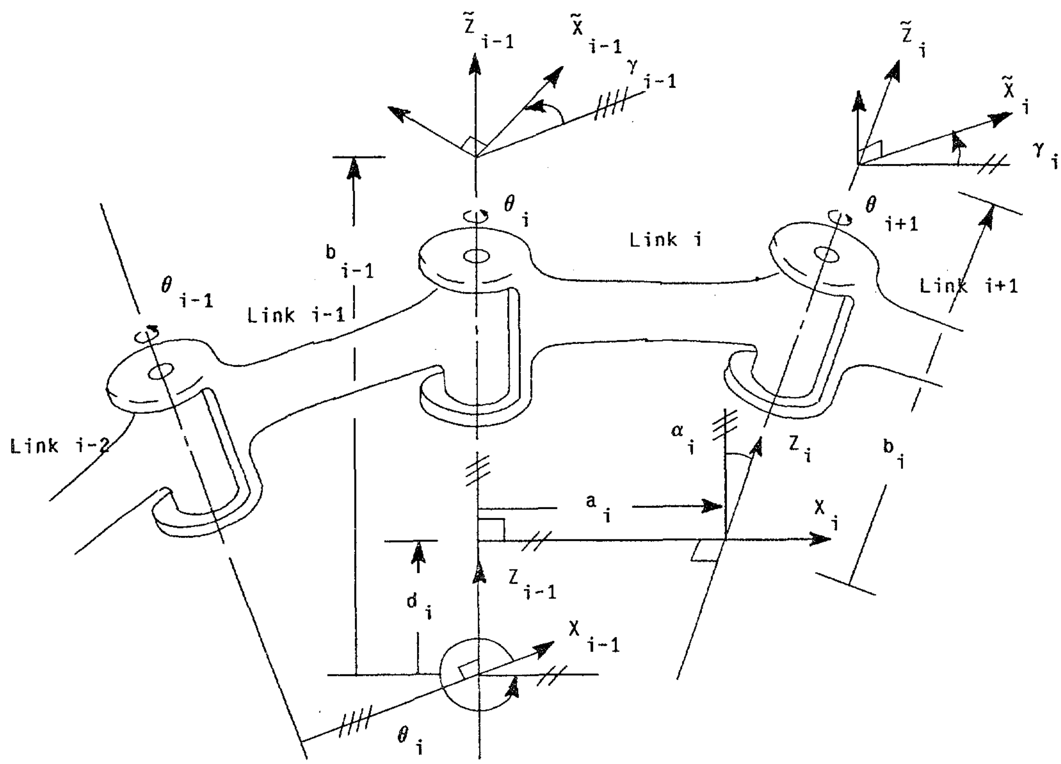

Figure 2: Definition of S-Model Kinematic Parameters

If joint i is *revolute*, β_i is a function of the joint position θ_i and the remaining five parameters $\bar{\mathbf{d}}_i$, $\bar{\mathbf{a}}_i$, $\bar{\alpha}_i$, γ_i, and $\mathbf{b}_i$ are constants. If joint i is *prismatic*, $\bar{\mathbf{d}}_i$ is a function of the joint position $\mathbf{d}_i$ and the remaining five parameters β_i, $\bar{\mathbf{a}}_i$, $\bar{\alpha}_i$, γ_i, and $\mathbf{b}_i$ are constants. We extract the four Denavit-Hartenberg parameters from the six S-Model parameters according to

$$\theta_i = \beta_i + \gamma_{i-1} \tag{9}$$

$$\mathbf{d}_i = \bar{\mathbf{d}}_i + \mathbf{b}_{i-1} \tag{10}$$

$$\mathbf{a}_i = \bar{\mathbf{a}}_i \tag{11}$$

$$\alpha_i = \bar{\alpha}_i \ . \tag{12}$$

In Section 3, we apply the following properties of the S-Model in (3) to develop our kinematic identification algorithm:

- The flexibility in assigning link coordinate frames (which leads to a simple, efficient, and accurate algorithm for identifying the location of the S-Model link coordinate frames $\mathcal{F}_i$ for $i=0,\ldots,n-1$); and
- The ability to extract the Denavit-Hartenberg model parameters from the S-Model parameters according to (9) - (12).

3 Kinematic Identification

3.1 Kinematic Features

The objective of *S-Model Identification* is to estimate the Denavit-Hartenberg kinematic parameters defined implicitly by *three* mechanical features (*centers-of-rotation* and *planes-of-rotation* for revolute joints, and *lines-of-translation* for prismatic joints). The locus of a point rotating about an axis is a circle lying in a plane. The normal to this plane is a vector which is parallel to the axis-of-rotation. Furthermore, the center of the circle is a point which lies on the axis-of-rotation. These two geometrical properties are the plane-of-rotation and center-of-rotation features. When joint *i-1* of a manipulator is rotated, any point which is fixed relative to the i^{th} link defines a plane-of-rotation and a center-of-rotation (under the assumption that the positions of joints 1 through *i-2* remain fixed). We associate the plane-of-rotation and center-of-rotation with the $(i\text{-}1)^{th}$ joint and the i^{th} link.

The line-of-translation is a feature of a *prismatic* joint. When a point is displaced linearly, its trajectory is a straight line which is parallel to the vector which indicates the direction of the displacement. For a manipulator, any point which is fixed relative to link *i* defines a line-of-translation when joint *i-1* is actuated (under the assumption that the positions of joints 1 through *i-2* remain fixed). These *three* features contain the essential information to formulate a complete kinematic model of a manipulator. Identification of these features is the first step in our S-Model identification algorithm.

3.2 Identification

Our S-Model identification algorithm proceeds through four steps:

- Feature Identification (in Section 3.2.1);
- Specification of the link coordinate frames (in Section 3.2.2);
- Computation of the S-Model parameters (in Section 3.2.3) and
- Extraction of the Denavit-Hartenberg parameters (in Section 3.2.4).

3.2.1 Feature Identification

We begin by applying recursive least-squares [3] to identify the three manipulator features. We estimate and reference the parameters which describe the position and orientation of the planes, points, and lines to a single coordinate frame, called the *sensor* frame, which must remain fixed relative to the base of the manipulator. Our systematic approach for collecting the measurement data insures that the identified features correspond to a fixed configuration of the manipulator which we define to be the *signature configuration*. The signature configuration is specified by the n measured joint positions $\mathbf{q}_i^s$ for $i=1,2,\ldots,n$. For expository convenience, we describe only the identification of the features of a *revolute* joint.

Identifying a plane-of-rotation and a center-of-rotation is straightforward. Let us imagine a target point fixed relative to link $i+1$. When revolute joint i is rotated, this target point traces a circle in space. The plane in which the circle lies is the *plane-of-rotation*, which is defined by the coefficients of the equation for the plane. The center of the traced circle, defined by its three-dimensional cartesian coordinates, is the *center-of-rotation*. In practice, the point may be a physical location on the $i+1^{st}$ link or a location of a point on a body which is attached rigidly to link $i+1$. The only constraint is that the cartesian position of the target can be measured by the sensor system which must remain *fixed* relative to the base of the manipulator throughout the identification procedure.

The coefficients of the plane are estimated from a recursive least-squares regression of m measured cartesian positions of the target along the circle. A natural criterion for plane-of-rotation parameter estimation is to minimize the sum of the squared *normal* errors between the measured positions and the estimated plane. While this estimate requires the solution of a *nonlinear* minimization problem, we have developed a sequential linear least-squares algorithm to approximate the solution. In practice, increasing m decreases the error in the estimated coefficients. Our approach leads to a robust algorithm which computes the coefficients (a_1, a_2, a_3) of the plane

$$\mathrm{z} = \mathrm{a}_1\mathrm{x} + \mathrm{a}_2\mathrm{y} + \mathrm{a}_3 \ . \tag{13}$$

To estimate the position of the center of the traced circle, the measured target positions are first projected normally onto the estimated plane-of-rotation. The projected data are then transformed to a coordinate frame whose X-Y plane is parallel to the estimated plane-of-rotation. After projection and transformation, the Z coordinates of the data equal a constant. The X and Y components of the data are then used to compute the coefficients (a_1, a_2, a_3) of the circle

$$\mathrm{w} = \mathrm{a}_1\mathrm{x} + \mathrm{a}_2\mathrm{y} + \mathrm{a}_3 \tag{14}$$

where w is the sum-of-squares of the X and Y components of the data.

The X and Y coordinates of the center of the circle are computed from the parameters $\mathbf{a}_1$, $\mathbf{a}_2$, and $\mathbf{a}_3$. The computed center of the circle is then transformed to the sensor coordinate frame by the inverse of the initial coordinate transformation. The coordinates of the center of the circle, which define the center-of-rotation for joint i, are denoted by $\vec{p}_{i,c} = [x_{i,c}, y_{i,c}, z_{i,c}]^T$.

3.2.2 Constructing $\bar{\mathbf{S}}_i$

In the second step, we apply the identified features to specify the location of the S-Model link coordinate frames which satisfy the S-Model convention (in Section 2.3). We compute the matrices

$$\bar{\mathbf{S}}_i = \mathbf{P}\cdot\mathbf{S}_i \qquad \text{for} \quad i=0,\ldots,n \tag{15}$$

to define the positions and orientations of the S-Model link coordinate frames with respect to the sensor coordinate frame. In (15), $\mathbf{P}$ is a constant homogeneous transformation matrix representing the spatial transformation from the sensor coordinate frame to the manipulator base coordinate frame $\mathcal{F}_0$. Placement of the sensor system relative to the manipulator is arbitrary, at least from the analytical point-of-view. The constant matrices $\bar{\mathbf{S}}_i$ describe the kinematics of the manipulator in the *signature configuration*. In Section 3.2.4, we generalize this constant model for the entire joint space.

In Figure 3, we illustrate construction of $\bar{\mathbf{S}}_i$ from the identified features. By definition

$$\bar{\mathbf{S}}_i = \begin{bmatrix} \vec{n}_i & \vec{o}_i & \vec{a}_i & \vec{p}_i \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

The unit direction vector $\vec{\mathbf{a}}_i$ is the unit normal vector to the estimated plane-of-rotation of the $i+1^{\text{th}}$ joint. The unit direction vector $\vec{n}_i$ is

$$\vec{\mathbf{n}}_i = \vec{\mathbf{p}}_{i+1,1} - \vec{\mathbf{p}}_{i+1,c}/|\vec{\mathbf{p}}_{i+1,1} - \vec{\mathbf{p}}_{i+1,c}| \tag{17}$$

where $\vec{\mathbf{p}}_{i+1,1}$ is the location of the first target position for joint $i+1$ (projected onto the plane-of-rotation). For convenience, we use the first target position. The Y axis direction vector $\vec{\mathbf{o}}_i$ is the cross product of the Z and X axis unit direction vectors $\vec{\mathbf{a}}_i$ and $\vec{\mathbf{n}}_i$, respectively.. The origin of the link i coordinate frame $\vec{\mathbf{p}}_i$ is the center-of-rotation $\vec{\mathbf{p}}_{i+1,c}$. When computing $\vec{\mathbf{a}}_i$ from the plane-of-rotation, the positive sense of $\vec{\mathbf{a}}_i$ must correspond with the positive sense of rotation of the $i+1^{\text{th}}$ joint.

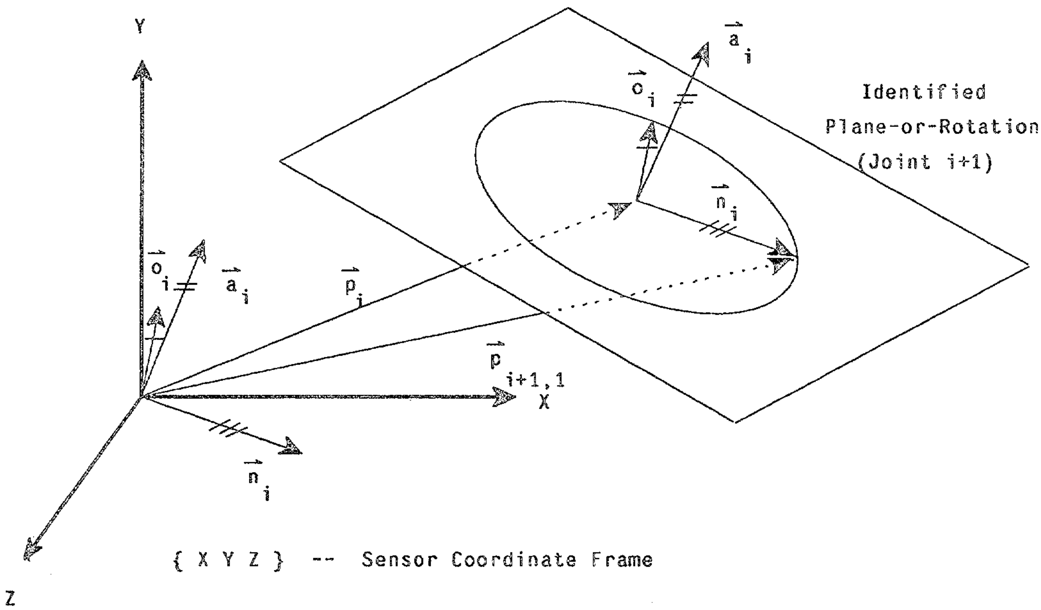

Figure 3: Coordinate Frame Construction for a Revolute Joint

3.2.3 S-Model Parameters

In the third step, we compute the transformation matrices B_i from the $\bar{S}_i$ matrices according to (3) and (15):

$$B_i \;=\; \bar{S}_{i-1}^{-1}\;\bar{S}_i \qquad \text{for} \quad i = 1, \ldots, n \;. \tag{18}$$

We then apply Paul's backward multiplication technique [6] to compute the six constant transformation matrix parameters β_i, $\bar{d}_i$, $\bar{a}_i$, $\bar{\alpha}_i$, γ_i, and b_i from B_i.

3.2.4 Denavit-Hartenberg Parameters

In the fourth and final step of our identification algorithm, we extract the Denavit-Hartenberg parameters to model the kinematics of the manipulator over the entire joint space. We recognize that the physical position of a joint, as measured by the joint encoders, may not coincide with the value of the joint position as defined in the Denavit-Hartenberg model. We denote the constant offset between these two measurements by q_i^{offset}. In terms of the signature configuration, q_i^{offset} is

$$q_i^{offset} \;=\; q_i^s \;-\; \bar{q}_i^s \tag{19}$$

where $\bar{q}_i^s$ is the encoder measured *signature configuration* position of joint *i* and q_i^s is the corresponding Denavit-Hartenberg position of joint *i* computed according to (9) for a revolute joint or (10) for a prismatic joint. The Denavit-Hartenberg model parameters, $q_i = \theta_i$ for a revolute joint and $q_i = d_i$ for a prismatic joint, defined in (1) are thus

$$q_i \;=\; \bar{q}_i \;+\; q_i^{offset} \tag{20}$$

The parameters θ_i and d_i are functions of the controllable and measureable position $\bar{q}_i$. With the exception of θ_1 or d_1, the remaining $(3 \cdot n) - 1$ constant Denavit-Hartenberg parameters are computed according to (9) - (12).

The Denavit-Hartenberg parameters θ_1 and d_1 are functions of the S-Model parameters γ_0 and b_0, respectively. In principle, γ_0 and b_0 can be computed from the elements of the matrix B_0. From (18), however, B_0 is undefined since neither $\bar{S}_{-1}$ nor equivalently S_{-1} is defined in our identification algorithm. Fortunately, θ_1 and d_1 are not essential for the kinematic modeling and control of manipulators. We circumvent the problem by modifying the model in (1). We replace the matrix T_n by $\bar{T}_n$ to represent the position and orientation of the n[th] link coordinate frame in terms of the *identified* S-Model base coordinate frame $\mathcal{F}_0$. To distinguish between the modified model and the original Denavit-Hartenberg model in (1), we express $\bar{T}_n$ as

$$\begin{aligned} \bar{T}_n \;&=\; \bar{A}_1 \cdot A_2 \cdot \ldots \cdot A_n \\ &=\; [\,\mathrm{Rot}(z\,,\,-\gamma_0)\,\mathrm{Trans}(0,0,-b_0)\,]\,T_n \end{aligned} \tag{21}$$

where the matrix $\bar{A}_1 \;=\; \bar{A}_1(\bar{q}_1)$ has the functional form of (2), and $A_i \;=\; A_i(\bar{q}_i)$ for $i = 2, \ldots, n$. We replace q_1 by

$$\bar{\theta}_1 \;=\; \bar{q}_1 \;+\; (\beta_1 \;-\; \bar{q}_1^s) \tag{22}$$

for a *revolute* joint and

$$d_1 \;=\; \bar{q}_1 \;+\; (\bar{d}_1 \;-\; \bar{q}_1^s) \tag{23}$$

for a *prismatic* joint. We call the model in (21) the *pseudo* Denavit-Hartenberg model. Even though the parameters γ_n and b_n are not required to formulate the pseudo Denavit-Hartenberg model in (21), we compute *all* of the S-Model parameters in the third step of our identification procedure. The parameters γ_n and $\mathbf{b}_n$ may be required to invert (21). In a subsequent paper, we will detail a general purpose algorithm to invert (21).

4 Implementation and Performance Evaluation

For kinematic identification, we place the manipulator in an initial configuration, the signature configuration. We position a three dimensional cartesian range sensor relative to the base of the manipulator. Measurements of the position of a target point for each link are gathered sequentially as each joint is actuated to define the features. From these measurements, we estimate the coefficients of the equations defining these features. The estimated coefficients are used (as outlined in Section 3.2.2) to compute the $\bar{S}_i$ matrices. For simplicity, we set $\bar{S}_n$ equal to $\bar{S}_{n-1}$. The n^{th} link S-Model coordinate frame is thus defined to coincide with the $(n-1)^{th}$ link S-Model coordinate frame when joint n is in the signature configuration position. We then compute the matrices $\mathbf{B}_i$ according to (18). The matrices $\mathbf{B}_i$ are functions of the S-Model parameters β_i, $\bar{\mathbf{d}}_i$, $\bar{\mathbf{a}}_i$, $\bar{\alpha}_i$, $\mathbf{b}_i$, and $\dot{\gamma}_i$ which are computed by Paul's backward multiplication method [6]. Finally, we extract the Denavit-Hartenberg parameters from the S-Model parameters according to (9) - (12). We then apply the identified pseudo Denavit-Hartenberg model to manipulator kinematic control.

S-Model identification can be used to obtain accurate models of the kinematic structures of all robotic manipulators with rigid links. We have implemented and applied our algorithm to identify the signatures of seven Unimation/Westinghouse Puma 560 robots. Such six degree-of-freedom robots with revolute joints are inherently sensitive to manufacturing errors, and it is a formidable task to characterize and compensate for these errors.

There are six components to our implementation:

- External range sensor system;
- Measurement collection and preprocessing software;
- Identification software;
- Signature-based control software;
- Position and orientation measuring apparatus for evaluation; and
- Performance test and evaluation software.

We use ultrasonic range sensors to measure the three dimensional cartesian positions of the target. The software which controls the Puma[1] 560 during the collection of the target positions and preprocesses these data is implemented on the Puma's Val II[2] controller. The software which estimates the feature parameters and computes the signature parameters was written in the C programming language to run under the Unix[3] operating system on a Digital Equipment Corporation VAX 780. The measurement data are thus collected by the VAL II controller and then uploaded to the VAX 780. Once identified, the arm signature is used to control the manipulator configuration in specified positioning tasks. A challenging task for analyzing the kinematic performance of a manipulator is the *three*-dimensional grid touching task. The Puma is initially taught three arbitrary points to define a workspace coordinate frame. During the task, the manipulator is programmed to position sequentially its end-effector at the vertices of a three dimensional grid defined with respect to the workspace coordinate frame. At each of the grid points, the actual position and orientation of the end-effector are measured. To obtain these measurements, we combined the Ranky sensor jig [7] and a Bridgeport milling machine. We mounted our sensor jig on the bed of a three-axis milling machine equipped with linear optical encoders. The jig can be displaced accurately over a large volume. The accuracy of our sensor jig is 0.001 inch in position and 0.01 degree in orientation. In the preliminary experimental results which follow, the grid size was 25 cm by 25 cm by 60 cm. Seperate software, residing on the VAX 780, processes the grid touching measurement data and tabulates indices of kinematic performance.

The performance of the identification algorithm and the subsequent increase in manipulator kinematic performance are illustrated for two Unimation/Westinghouse Puma 560 robots. The Denavit-Hartenberg parameters of the ideal kinematic model of a Puma 560 *prior to manufacture* are listed in Table 1 [8]. In Table 2, we list the pseudo Denavit-Hartenberg parameters for the two Puma 560 robots identified by the S-Model identification algorithm. The estimated values in Table 2 illustrate the sensitivity of the Denavit-Hartenberg parameters to small mechanical errors in the physical manipulator. For instance, the values of $\mathbf{d}_2$ and $\mathbf{d}_3$ in Table 2 indicate that the joint 2 and joint 3 axes are slightly out-of-parallel. The dramatic differences between the identifed parameters and the ideal parameters arise from small manufacturing errors.

The grid touching task described above was performed twice by each of the two robots: with the ideal parameters and with the arm signature parameters. In Table 3, we summarize four measured indices by which we evaluate manipulator kinematic performance. This enhanced performance, along with the comparable improvement of the other five Puma 560s, indicate that the S-Model identification algorithm is a viable approach to identify the actual kinematics of manipulators and enhance their kinematic performance.

Unimate Puma 560 Denavit-Hartenberg Kinematic Parameters				
Link	Variable	d (cm)	a (cm)	α (deg)
1	θ_1	0.0	0.0	-90.0
2	θ_2	14.909	43.18	0.0
3	θ_3	0.0	-2.032	90.0
4	θ_4	43.07	0.0	-90.0
5	θ_5	0.0	0.0	90.0
6	θ_6	0.0	0.0	0.0

Table 1: Ideal Kinematic Parameters

[1] Puma is a trademark of the Westinghouse/Unimation Corporation.

[2] Val II is a trademark of the Westinghouse/Unimation Corporation.

[3] Unix is a trademark of AT&T Bell Laboratories, NJ

Robot 1					
Link	Variable	θ_{offset} (deg)	d (cm)	a (cm)	α (deg)
1	θ_1	110.308	-20.678	-0.005	-90.066
2	θ_2	79.977	11,210.213	7.607	-0.217
3	θ_3	-79.483	-11,194.978	-2.039	90.519
4	θ_4	-0.296	43.278	-0.001	-90.007
5	θ_5	-0.497	-0.018	0.002	89.990
6	θ_6	-89.081	10.300	0.000	0.000

Robot 2					
Link	Variable	θ_{offset} (deg)	d (cm)	a (cm)	α (deg)
1	θ_1	110.849	-20.428	-0.023	-90.013
2	θ_2	63.467	2,783.580	19.084	-0.803
3	θ_3	-63.276	-2,768.574	-2.028	90.134
4	θ_4	0.328	43.299	-0.009	-89.991
5	θ_5	-0.126	-0.054	0.007	89.969
6	θ_6	-89.517	10.273	0.000	0.000

Table 2: Identified Arm Signature Parameters

Puma 560 Experimental Performance Improvement [Performance with Arm Signature/Performance without Arm Signature]		
Description	Robot 1	Robot 2
Maximum Normal Deviation from a Line	4.6	7.7
Deviation from Perpendicular between Grid Lines	20.0	14.3
Maximum Radial Deviation of Grid Point from Desired Location	9.1	4.8
Maximum Orientational Deviation of End-Effector from the Desired Orientation	33.3	4.2

Table 3: Puma 560 Kinematic Performance Summary

5 Conclusion

In this paper, we formulate the *S-Model* to describe the kinematic structure of robotic manipulators and introduce *three* features to characterize their kinematics. The S-Model is the analytical foundation of our arm signature identification algorithm. The readily identifiable features are used to estimate the $6 \cdot n$ S-Model parameters from which we extract the Denavit-Hartenberg parameters. We apply these parameters to synthesize a kinematic control algorithm to improve manipulator kinematic performance. A sequential linear least-squares algorithm is developed to estimate the feature parameters from the measured positions of a target mounted on the manipulator links. The sensor system to acquire the measurement data and its spatial placement are independent of the manipulator thus eliminating the need for elaborate and costly fixturing. Furthermore, positioning of the manipulator during data collection is totally automated. Finally, we demonstrate the viability and practicality of our approach and highlight the improved kinematic performance of seven Puma 560 robots through hardware implementation.

6 Acknowledgments

This work is supported by Westinghouse/Unimation, and the Robotics Institute and Department of Electrical and Computer Engineering of Carnegie-Mellon University. We thank Richard Casler and Rajan Penkar of Westinghouse/Unimation and Dr. Eugene Bartel of Carniegie-Mellon University for their support and contributions to our research.

References

[1] Chen, J., Wang, C. B., and Yang, C. S.
Robot Positioning Accuracy Improvement through Kinematic Parameter Identification.
In *the Proceedings of the Third Canadian CAD/CAM in Robotics Conference*, pages 4.7-4.12. June, 1984.
Toronto, Canada.

[2] Denavit, J. and Hartenberg, R. S.
A Kinematic Notation for Lower-Pair Mechanisms Based on Matrices.
Journal of Applied Mechanics 77(2):215-221, June, 1955.

[3] Goodwin, G.C. and Sin, K. S.
Adaptive Filtering, Prediction, and Control.
Prentice-Hall, Englewood Cliffs, NJ, 1984.

[4] Hayati, S. and Mirmirani, M.
Puma 600 Robot Arm Geometric Calibration.
In *The First IEEE International Conference on Robotics.* March 13, 1984.
Atlanta, GA.

[5] Mooring, B. W. and Tang, G. R.
An Improved Method for Identifying the Kinematic Parameters in a Six Axis Robot.
In *the Proceedings of the International Computers in Engineering Conference*, pages 79-84. ASME, New York, August, 1984.
Las Vegas, NV.

[6] Paul, R. P.
Robot Manipulators: Mathematics, Programming, and Control.
The MIT Press, Cambridge, MA, 1981.

[7] Ranky, P. G.
Test Method and Software for Robot Qualification.
The Industrial Robot 11(2):111-115, June, 1984.

[8] *Unimate Puma Mark II 500 Series Equipment Manual 398U1*
Unimation Incorporated, Danbury, CN, 1985.

[9] Whitney, D. E., and Lozinski, C. A.
Industrial Robot Calibration Methods and Results.
In *the Proceedings of the International Computers in Engineering Conference*, pages 92-100. ASME, New York, August, 1984.
Las Vegas, NV.