Statistical Performance Evaluation of the S-Model Arm Signature Identification Technique

Henry W. Stone Arthur C. Sanderson

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

The S-Model arm signature identification algorithm [2,3,4,5] is a general technique for identifying the kinematic parameters of any "n" degree-of-freedom robotic manipulator with rigid links. This technique provides a means to reduce the modeling errors between the actual kinematics of a robot and the kinematic model used to control the end-effector position and orientation. In this approach, the nominal control model whose parameters are obtained from the robot's mechanical design specifications, and which do not account for the presense of random manufacturing errors, is replaced by the identified arm signature model. In [3,4], the S-Model identification algorithm has been applied to identify the kinematic parameters and improve the kinematic performance of seven Unimation/Westinghouse Puma 560 robots.

In this paper, we apply Monte-Carlo simulation techniques to gain further insight into the relationship between manufacturing errors and the performance of a robot using either the design model or arm signature model for control. In conventional design-model robot control, manufacturing errors contribute most to robot positioning errors. Thus, we relate the statistical parameters which characterize a robot's positioning accuracy to the statistical parameters which characterize the manufacturing error probability distribution functions. In arm signature-based robot control (S-Model), the correct arm signature model eliminates kinematic errors due to manufacturing. In this case, robot performance is limited by sensor errors which contribute to inaccuracy of the identified arm signature model. The relationship between the statistical parameters which characterize a robot's positioning accuracy to the statistical parameters which characterize the sensor performance is presented. Finally, we analyze and quantify the requirements of an arm signature identification system in terms of the underlying sensor performance.

1 Introduction

In this paper, we analyze and evaluate the statistical performance of the S-Model Identification Technique and compare it to the statistical performance of the widely used Design Model control approach. In both cases, Monte-Carlo simulation techniques are applied to derive the underlying statistical relationships. In conventional *design model robot control*, the robot design model, whose parameters are obtained from the robot's mechanical design specifications and which do not account for the presence of random manufacturing errors, is used as a basis for kinematic control. In this case, manufacturing errors contribute most to robot positioning errors. In *arm signature-based robot control* (S-Model) where the robot design model is replaced by the identified arm signature model, the correct arm signature model eliminates kinematic errors due to manufacturing. In this case, robot kinematic performance is limited by sensor errors which contribute to inaccuracy of the identified arm signature model.

Experimental studies of D-Model and S-Model control are presented in [3,4,5]. In these studies, the S-Model control showed consistent performance improvement, and demonstrated accuracy approaching limits predicted by the joint encoder resolution. In this paper, we analyze the origin of this improvement, and quantify the requirement of an arm signature system in terms of the underlying sensor performance.

D-Model control performance depends upon the manufacturing process which determines the actual geometry of the robot. Since manufacturing errors vary randomly from one robot to another, the performance or an arbitrarily chosen robot has a random component. The statistical parameters which characterize a robot's positioning accuracy are related to the statistical parameters which characterize the manufacturing error probability distribution functions. Unfortunately, the complexity of robot kinematics prevents us from analytically deriving this relationship. Thus, we applied Monte-Carlo simulation techniques to gain insight into the relationship between manufacturing errors and the performance of a design model controller. A kinematic model of a Puma 560 robot which directly incorporates manufacturing errors as parameters was developed for this analysis and is described in Section 3. The results of the analysis are presented in Section 4 and bring to light some of the major disadvantages of the design model control approach.

The performance of S-Model control is dictated by the presence of errors in the arm signature identification process and not by the presence of manufacturing errors. In this case, errors are introduced into the identification process by the sensor system and must be analyzed to determine if the performance of the robot would improve relative to the performance obtained from using D-Model control. In section 4 we analyze the relationship between target measurement errors and the performance of S-Model control using simulation methods.

This paper is organized as follows. Section 2 provides a brief review of the S-Model identification algorithm and our prototype identification system [3]. In Section 3 we describe the structure of our Monte-Carlo simulator. Then, in Sections 4.1 and 4.2, we explore the relationship between the performance of the D-Model controller and the two types of manufacturing errors referred to as *encoder calibration errors* and *machining and assembly errors*. Sections 4.3 - 4.5 discuss the relationships between the performance of the S-Model controller and the target measurement accuracy, the number of measurements, and the length of the target radii, respectively. Finally, in Section 5 the issue as to whether or not S-Model control improves robot end-effector positioning accuracy is resolved.

2 Background

In [2], we introduced a new kinematic model for describing robot kinematics called the S-Model. This model was developed specifically to facilitate kinematic parameter identification. The identification algorithm which was also developed and is based upon the properties of this model and the observed intrinsic properties of mechanical joints is called the *S-Model Identification Algorithm*. This section presents a brief description of the model, the identification algorithm and a prototype system used to identify the signatures of seven Puma 560 robots. Our intent is simply to establish the definition of terms used later in this paper. Detailed descriptions of the above can be found in [2,3,4,5].

In the S-Model, the matrix

$$S_n = B_1 \cdot B_2 \cdot \ldots \cdot B_n \tag{1}$$

defines the position and orientation of a coordinate frame fixed relative to the last link of a manipulator with respect to a coordinate frame fixed relative to the base link. The general transformation matrices

$$B_i = Rot(z,\beta_i)Trans(0.0,0.0,\overline{d}_i)Trans(\overline{a}_i,0.0,0.0)Rot(x,\overline{\alpha}_i) \\ Rot(z,\gamma_i)Trans(0.0,0.0,b_i) \quad (2)$$

in (1) describe the relative transformation between cartesian coordinate frames δ_{i-1} and δ_i fixed to links $i-1$ and i, respectively. Each matrix B_i is a function of the six kinematic parameters, β_i, $\overline{d}_i$, $\overline{a}_i$, $\overline{\alpha}_i$, γ_i and b_i. The parameter β_i is a function of the joint i position for revolute joints and $\overline{d}_i$ is a function of the joint i position for prismatic joints. As a result of its structure, the S-Model possesses three important features which make it directly applicable to kinematic identification. First, there is a considerable amount of flexibility in assigning the locations of the link coordinate frames. Second, the locations of the link coordinate frames are independent of the locations of the other link coordinate frames. And third, the Denavit-Hartenberg parameters can be easily extracted from the S-Model parameters.

The objective of S-Model identification is to estimate the S-Model kinematic parameters from a set of $2n_r+n_p$ kinematic/mechanical features inherent to the manipulator, where n_r is the number of revolute joints and n_p is the number of prismatic joints. The kinematic features of a revolute joint are the *plane-of-rotation* and a *center-of-rotation*, and the kinematic feature of a prismatic joint is a *line-of-translation*. In particular, it is recognized that the locus of point rotating about an axis is a circle which lies in plane and that the vector normal to this plane is parallel to the axis of rotation. In addition, the center of the circle is a point which lies on the axis of rotation. When joint $i-1$ is rotated, any point which is fixed relative to the i^{th} link defines a plane-of-rotation and a circle-of-rotation provided that the positions of joints 1 through $i-2$ remain fixed. A similar scenario applies to prismatic joints. A set of such kinematic features contains the essential information required to formulate a complete kinematic model since together the features determine the relative locations of the joint axes.

The S-Model identification algorithm consists of four steps in which the overall identification problem is separated into a set of independent, less complex identification problems.

In the first step, *feature identification* the kinematic features are identified from measurements of the Cartesian position of targets physically attached to the robot's links. The positions of these targets, relative to an independent fixed coordinate frame (i.e., the sensor frame) vary as the manipulator changes configuration. During data collection the manipulator is programmed to move through a sequence of joint configurations. At each configuration the position of a target is measured. The kinematic features are described analytically by an algebraic equation, and the coefficients of these equations are the feature parameters which are estimated.

The second step in the identification algorithm is *link coordinate frame specification*. If the manipulator configuration is known in terms of the joints positions a set of link coordinate frames which satisfy the S-Model convention [2,4,5] may be defined. By collecting target measurements and identifying the parameters of the kinematic features which correspond to this same manipulator configuration, we can readily establish a valid set of S-Model link coordinate frames. Then, we can apply the estimated feature parameters to construct and evaluate the elements of the matrices $\overline{S}_i$ where $\overline{S}_i = P * S_i$ and P is a constant yet unknown transformation matrix representing the spatial transformation between the sensor coordinate frame and the Link 0 coordinate frame.

The third step, *S-Model parameter computation* applies the application of the inverse kinematic parameter relationships for β_i, $\overline{d}_i$, $\overline{a}_i$, $\overline{\alpha}_i$, γ_i and b_i developed in [4]. The elements of the general transformation matrices B_i, which are the arguments to these inverse relatiohships, are a function of the transformation matrices $\overline{S}_i$ determined in the previous step. Identifying the S-Model parameters requires straightforward numerical evaluation.

In the fourth and final step, *Denavit-Hartenberg parameter extraction* relationships developed in [4] are applied to determine the manipulators Denavit-Hartenberg kinematic parameters from the identified S-Model parameters.

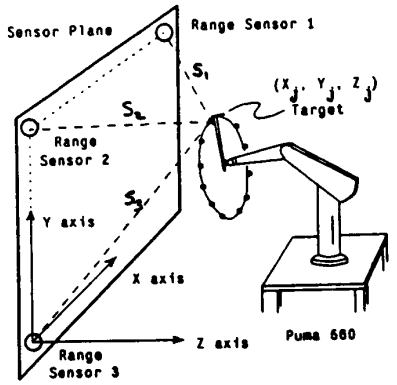

Figure 1: Measurement Collection Scenario

Figure 1 illustrates the basic scenario under which measurements are obtained for the identification algorithm using our prototype system. The figure depicts a Puma 560 robot sequencing the position of joint 6 in order to generate a plane-of-rotation and center-of-rotation for joint 6. (Note that a fixture has been attached to Link 6 to support the target at an appropriate location). N_i measurements of the i^{th} link's target positions denoted by $\vec{P}_j = (x_j, y_j, z_j)$ for $j = 1, 2 \ldots N_i$ are computed based upon the explicit measurments of the three slant ranges S_1, S_2, and S_3. In our system the target (source) is an ultrasonic sparker and the detectors are ultrasonic microphones. Standard triangulation algorithms are used to perform the computations and are a function of the range sensor spatial separations. In addition to controlling the identifier parameters N_i for $i = 1, 2, \ldots 6$ one also has control over the nominal radius R_i of the i^{th} target point relative to the i^{th} joint axis. Increasing R_i with in the limitations of the sensor system workspace tends to increase feature estimate accuracy. Naturally, both the number of measurements N_i and the nominal target radii R_i should be used to reduce the effects of the slant range measurement noise upon feature estimate accuracy and thus the accuracy of the identified signatures.

3 A Monte-Carlo Simulator

The performance evaluation of the D-Model and S-Model controllers is complicated by the nonlinear robot kinematics, the nonlinearities of the S-Model identification algorithm, and the presence of error sources. Random errors are introduced into both the manufacturing and identification processes. This reality forced us to develop a Monte-Carlo simulator to conduct a statistical evaluation. Because of the variability of robot designs, development of a general-purpose Monte-Carlo simulator was impractical. Thus, to complement our hardware experimentation [3,4], we have evaluated the kinematic performance of the Puma 560 robot with both design model based control and signature-based control. In doing so, we have established a methodology to evaluate the kinematic performance of all robots.

In our simulator we used a three-dimensional grid touching task to evaluate the kinematic performance of the Puma 560 with alternative controller designs. The grid contains twelve vertices, labeled 1 through 12, whose approximate positions, relative to the robot, are depicted in Figure 2. The desired positions of the vertices, measured with respect to the base coordinates of an ideal Puma 560 are listed in Table 1. The desired orientation of the end-effector is the same for all twelve vertices. Vertices 1 through 8 define four lines parallel to the X axis (Lines 1, 4, 5, and 8) each 60.0 cm long, four lines parallel to the Y axis (Lines 2, 3, 6, and 7) each 30.0 cm long, and four lines parallel to the Z axis (Lines

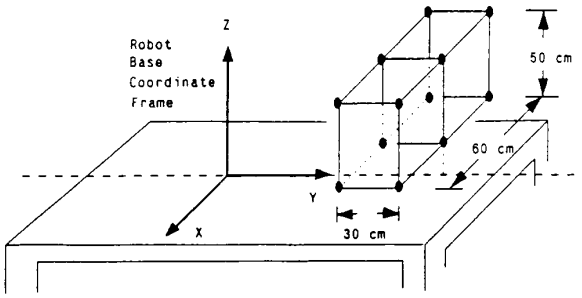

Figure 2: Definition and Location of Points in the Simulated Three-Dimensional Grid Touching Task

Point	Cartesian Position		
	X (cm)	Y (cm)	Z (cm)
1	-30.0	40.0	30.0
2	30.0	40.0	30.0
3	-30.0	70.0	30.0
4	30.0	70.0	30.0
5	-30.0	40.0	80.0
6	30.0	40.0	80.0
7	-30.0	70.0	80.0
8	30.0	70.0	80.0
9	0.0	40.0	30.0
10	0.0	70.0	30.0
11	0.0	40.0	80.0
12	0.0	70.0	80.0

Table 1: Positions of Three-Dimensional Grid Vertices

9, 10, 11, and 12) each 50.0 cm long. The four vertices 9 through 12 are the midpoints of lines 1, 4, 5, and 8, respectively.

Six types of positioning and orienting errors are used to measure and evaluate robot performance.

1. *Linear Displacement Errors* – These errors are defined as the difference between the actual and desired lengths of various grid lines. Sixteen linear displacement errors are considered, twelve corresponding to the errors between the actual and desired lengths of Lines 1 through 12 and the four corresponding to the errors between the actual and desired lengths of the diagonal lines defined by Vertices, 1 and 4, 2 and 3, 5 and 8, and 6 and 7.

2. *Normal Deviation Errors* – These errors are defined as the normal distance between the end-effector's position at one of the midpoint vertices and the line joining the two actual positions of the end-effector at the lines' endpoints. These errors represent a measure of line straightness. For each of the lines 1, 4, 5, and 8, and (normal deviation error) is computed.

3. *Radial Displacement Errors* – These errors are defined as the radial distances between the actual and desired positions of the end-effector. Errors at Vertices 1, 4, 5, and 8 are computed.

4. *Absolute Orientational Errors* – These errors are defined as the angular deviation between the actual and desired orientations of the end-effector. Absolute orientational errors are computed at Vertices 1, 4, 5, and 8.

5. *Relative Orientational Errors* – Each of these errors are defined as the angular deviation between the actual orientation of the end-effector at a specified vertex and the orientation of the end-effector at the grid origin (i.e., Vertex 1).

6. *Line Perpendicularity Errors* – The *X-Y Line Perpendicularity Error* is the angular deviation between Line 1 and Line 3, minus 90.0 degrees. Similarly, the angular deviations between Lines 3 and 8 and Lines 8 and 1 are the *Y-Z* and *Z-X Line Perpendicularity Errors*, respectively.

A total of thirty-four positioning and orienting errors are computed during each experiment (i.e., simulated event) within a simulator run in order to measure the robot's kinematic performance. Each error represents an index of performance and each simulation consists of 500 experiments. Following the completion of each simulation, we compute the mean, variance, and standard deviation of each performance index. Different sets of input parameters are specified in each simulation. To analyze the effects of manufacturing errors upon the performance of the D-Model controller, the input parameters are the manufacturing error variances while to analyze the performance of the S-Model controller, the input parameters are the nominal target radii R_i, number of points per circle N_i, and the sensor measurement noise standard deviation σ_s.

The performance of the D-Model controller is a direct reflection of the manufacturing errors. To simulate the performance, we have developed a kinematic simulator model of the Puma 560. The model

$$T_6^* = K_0 \cdot K_1 \cdot K_2 \cdot K_3 \cdot K_4 \cdot K_5 \cdot K_6 \tag{3}$$

where

$$\begin{aligned} K_0 &= Trans(\epsilon_1,\epsilon_2,\epsilon_3)Rot(x,\epsilon_4)Rot(y,\epsilon_5)Trans(0.0,0.0,x_1) \\ &\quad Rot(x,\epsilon_6)Rot(y,\epsilon_7)Rot(z,\epsilon_8) && (4) \\ K_1 &= Rot(z,\theta_1)Trans(0.0,0.0,\epsilon_9)Rot(x,\epsilon_{10})Rot(y,\epsilon_{11}) \\ &\quad Trans(\epsilon_{12},0.0,x_2)Rot(x,\epsilon_{13}+x_3)Trans(0.0,0.0,x_4) \\ &\quad Rot(z,\epsilon_{14}) && (5) \\ K_2 &= Rot(x,\theta_2)Trans(0.0,0.0,\epsilon_{15}+x_5)Rot(x,\epsilon_{16})Rot(y,\epsilon_{17}) \\ &\quad Trans(\epsilon_{18}+x_6,0.0,0.0)Rot(x,\epsilon_{19})Rot(y,\epsilon_{20})Rot(z,\epsilon_{21}) && (6) \\ K_3 &= Rot(z,\theta_3)Trans(0.0,0.0,\epsilon_{22})Rot(x,\epsilon_{23})Rot(y,\epsilon_{24}) \\ &\quad Trans(x_7+\epsilon_{25},x_8+\epsilon_{26},x_9+\epsilon_{27})Rot(x,x_{10}+\epsilon_{28}) \\ &\quad Rot(y,\epsilon_{29})Rot(z,\epsilon_{30}) && (7) \\ K_4 &= Rot(z,\theta_4)Trans(\epsilon_{31},0.0,x_{11}+\epsilon_{32})Rot(x,x_{12}+\epsilon_{33}) \\ &\quad Rot(z,\epsilon_{34}) && (8) \\ K_5 &= Rot(z,\theta_5)Trans(0.0,0.0,\epsilon_{35})Rot(x,x_{13}+\epsilon_{36}) \\ &\quad Trans(\epsilon_{37},0.0,x_{14})Rot(z,\epsilon_{38}) && (9) \\ K_6 &= Rot(z,\theta_6)Trans(0.0,0.0,x_{15}+\epsilon_{39}) && (10) \end{aligned}$$

incorporates both the nominal mechanical design specifications and the manufacturing errors as parameters. The homogenous transformation matrix T_6^* defines the position and orientation of a coordinate frame fixed relative to the end-effector with respect to a coordinate frame fixed relative to the base. The parameters, x_1, x_2, ..., x_{15}, represent the nominal mechanical design specifications while the parameters ϵ_1, ϵ_2, ..., ϵ_{39}, represent the manufacturing errors. The angular positions of joints 1 through 6, as measured by the respective joint encoders, are denoted by θ_1 through θ_6, respectively. The transformations comprising the matrix K_0 describe the actual physical structure of the base link while the matrices K_1, K_2, ..., K_6 describe the actual physical structure of Links 1, 2, ..., 6, respectively, for a particular Puma 560 robot. A deriviation of the model (3) is presented in [4,5].

The values of the nominal design parameters for the Puma 560, x_1 - x_{15}, are listed in Table 2. When the manufacturing errors are zero, the model (3) is nearly equivalent to the standard Denavit-Hartenberg model of the Puma 560 [4]. The difference is caused by the fact that, in

(3), we consider the base frame to be located at the lower end of Link 0 (i.e., coincident with the surface to which the robot would normally be mounted).

The values listed in Table 2 where obtained from a cursory examination of the machanical structure of a Puma 560 and not from

Parameter	Value (cm)	Value (deg)
x_1	58.500	—
x_2	8.900	—
x_3	—	-90.0
x_4	17.750	—
x_5	2.500	—
x_6	43.180	—
x_7	-2.032	—
x_8	-33.320	—
x_9	-5.341	—
x_{10}	—	90.0
x_{11}	10.050	—
x_{12}	—	-90.0
x_{13}	—	90.0
x_{14}	4.445	—
x_{15}	1.270	—

Table 2: Nominal Design Kinematic Parameters for a Puma 560.

an exhaustive study of the actual manufacturing process. Such a study was beyond the scope of our intended investigation. It is believed that the model (3) is sufficiently complex and realistic enough to provide insight into the expected performance of a Puma 560 whose kinematics are affected by random manufacturing errors.

In simulation, the manufacturing errors are generated using standard random number generators. Gaussian distributions are used to model the variations in manufacturing errors from one robot to another. It is assumed that all manufacturing errors have zero mean.

The manufacturing errors $\epsilon_1, \epsilon_2, \ldots, \epsilon_{39}$ are divided into three catagories; *positional errors*, *orientational errors*, and *encoder calibration errors*. For convenience, it is assumed that errors within a category have the same variance. The variance of the errors in the *positional error* catagory is measured in cm^2 and is denoted by σ_p^2. The variances of the errors in the remaining two catagories are measured in deg^2 and are denoted by σ_o^2 and σ_e^2, respectively. Our experience with Puma 560 robots suggests that the orientational errors introduced during the calibration of the joint encoders are significantly greater than the orientational errors committed during machining and assembly thus the need for two separate orientational error variances.

Simulating the performance of the S-Model controller is complicated by the need to simulate the identification process. Arm signature identification in combination with signature-based contron proceeds through a sequence of five steps. Table 3 lists the inputs, the outputs, and the potential sources of errors for each of these steps. The table indicates how the input errors propagate through the identification algorithm to produce end-effector positioning and orienting errors. The outputs of a step are affected by both the errors of the step and the errors of previous steps. The error sources listed in Steps 1 and 2 are specific to the ultrasonic sensor system used in our prototype system [3]. In practice, by interchanging simulator modules, alternative sensor systems can be analyzed.

Since it would be impractical to simulate all of the errors listed in Table 3 and interpret the results, our primary objective has been to evaluate on the effect of sensor errors (i.e., the errors which affect the accuracy of the target measurements) on the performance of the S-Model Controller. In our simulation of the Puma 560, we have considered only the sensor errors which most strongly affect the measurement of the target loci. These include slant range errors and sensor system calibration errors.

The S-Model identification algorithm is applied, in simulation, to identify the kinematic parameters of a perfectly manufactured Puma 560 robot in the presence of slant range errors and sensor system calibration errors. The assumption that the actual robot has no manufacturing errors merely simplified the simulator design task. Since the S-Model identification algorithm is a general method and does not require a priori knowledge as to the nominal kinematic structure of the robot, the statistical performance of a S-Model Controller for a perfectly manufactured Puma 560 will be identifical to the performance of an S-Model Controller for a Puma 560 with an arbitrary set of manufacturing errors.

Step	Description	Inputs	Outputs	Sources of Error
1	Sensor Model	Actual Target Range	Filtered Target Range	Temperature, Acoustic Noise, Numerical
2	Sensor System Model	Filtered Target Ranges	Cartesian Target Position	Sensor Misalignment, Numerical
3	Feature Model	Cartesian Target Positions	Feature Parameter Vector	Joint Wobble, Link Compliance, Gear Backlash, Transmission Compliance
4	S-Model	Feature Vectors	S-Model Parameters	Numerical
5	D-H Model	S-Model Parameters	Denavit-Hartenberg Model Parameters	Numerical
6	Control Model	Denavit-Hartenberg Model Parameters Desired End-Effector Position	Actual End-Effector Position	Steady-State Joint Position Control Errors, Encoder Resolution, Link Compliance, Backlash Numerical

Table 3: Propagation of Identification and Control Errors.

To initialize the simulator, a variety of input parameters must be specified such as the true locations of the target points with respect to the links, and spatial transformation between the sensor coordinate frame and the robot base coordinate frame. A list of the input parameters and the values used in our simulation experiments can be found in [4,5]. The values coincide with our hardware implementation of the S-Model identification algorithm and our prototype system.

The first computational task performed by the simulator is to use the Puma 560 design model to compute the actual locations of the target points with respect to the sensor coordinate frame. Then, the actual slant ranges are computed based upon the known positions of the microphones. Next, zero mean gaussian noise is added to each of the slant ranges. The standard deviation of this noise depends upon the range. The linear approximation

$$\sigma = 9.6432e - 5X + 0.003 \tag{11}$$

models the relationship between the noise standard deviation, σ, and the range, X. In (11), σ and X are measured in centimeters. The corrupted slant ranges are also quantized to two decimal places to simulate the limited resolution of the GP-8-3D ultrasonic digitizer [1]. Using these corrupted slatn ranges, the S-Model identification algorithm is applied to determine the robot's arm signature. This procedure is repeated 500 times producing 500 arm signatures. The Newton-Raphson algorithm [4] is then applied to control the actual Puma 560 robot in performing the grid touching task based upon each of the identified signatures. Finally, the means, variances, and standard deviations of the thirty four performance indices are computed and tabulated for later analysis. Prior to its use in our analysis a series of specially designed experiments were conducted to validate the behavior and numerical values generated by the simulator. A detailed description of the validation procedure and results can be found in [4,5]. The results therein confirm that the simulator produces realistic and reliable data.

4 Results

Our analysis of the statistical effect of manufacturing errors and identification errors upon end-effector positioning accuracy is divided into five parts. In Section 4.1, we consider the effects of encoder calibration errors upo the performance of the D-Model Controller. The effects of machining and assembly errors upon the performance of the D-Model Controller are discussed in Section 4.2. Sections 4.3 - 4.5 discuss the relationship between the performance of the S-Model Controller and sensor system accuracy, the number of measurements, and the target radii, respectively.

4.1 Encoder Calibration Errors

The procedure for calibrating the joint encoders of a Puma 560 can often lead to significant encoder calibration errors. These errors result in fixed biases between the actual and measured positions of the joints. Five simulations were conducted using model (3) to evaluate the effect of random encoder calibration errors upon robot performance. In these simulations, the positional error standard devaition and orientational error standard deviation were $\sigma_o = 0.01cm$ and $\sigma_p = 0.01deg$, respectively. The standard deviation of the encoder calibration error σ_e was varied from 0.1 deg to 0.5 deg. The statistical variations in the 34 performance indices of 500 simulated robots are plotted as function of σ_e in Appendix G, in [4,5]. (Due to space limitations, it is impossible to present all of these results in this article).

The standard deviations of all 34 performance indices are seen to increase linearly as the standard deviation of the encoder calibration errors increase. To illustrate, the radial position errors are Vertices 1, 4, 5, and 8 are plotted as a function of σ_e in Figure 3. From Figure 3, it is also observed that the rate at which different index standard deviations increase varies. For instance, the standard deviation of the Vertex 1 radial position error increases 1.4 times faster than that at Vertex 4. It is believed that the variability in the rate at which the

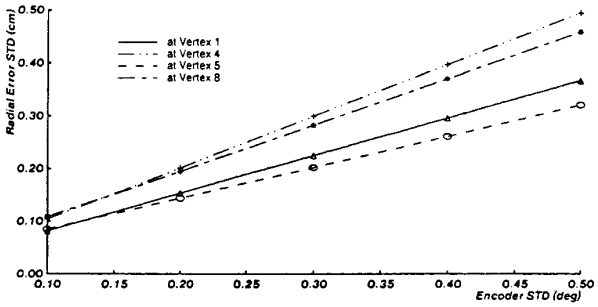

Figure 3: Radial Position Error Standard Deviations as a Function of σ_e.

performance index standard deviations increase as the encoder calibration error standard deviation increases is partly due to the variability in robot performance over the workspace. Robots with revolute joints, particularly those with revolute joints in the positioning system (i.e., the first three joints), tend to amplify joint positioning errors. Roughly speaking, the amplification factors are the effective radii between the axes where the errors occur and the end-effector. Thus, as a robot extends its reach the various amplification factors tend to increase. In design model control, encoder calibration errors translate into joint positioning errors. Thus, we would expect robot kinematic performance to decrease as the desired end-effector position approaches the outer limits of the robot's workspace. Figure 3 and the remaining plots in [?] clearly support this hypothesis. The standard deviation of the radial error at Vertices 4 and 8 are .299 cm and .282 cm, respectively, while the standard deviation of the radial error at Vertices 1 and 5 are .225 cm and .202 cm, respectively – when $\sigma_i^2 = 0.3deg$. From Table 1 we see that Vertices 4 and 8 lie near the maximum reach of the Puma 560 while Vertices 1 and 5 lie relatively close to the base of the robot.

There is another important feature of the effect of manufacturing errors upon performance which is not apparent from the plots. Even though the simulated manufacturing errors have zero mean, the means of the performance indices are nonzero. In fact for certain performance indices, the indices' mean values may be larger than their standard deviations. Furthermore in some cases, the index means vary linearly with the encoder calibration error standard deviation.

The mean value of the Line 11 link length error decreases from 0.0268 cm to 0.0038 cm as σ_e increases from 0.1 deg to 1.5 deg. Other index mean values may even change sign. Robot kinematics, especially those of the Puma 560, are highly nonlinear and coupled. As is often the case for nonlinear systems, the expected value of the system output, in this case robot performance, is both a function of the mean and standard deviation of the input noise. These results are in direct contradiction to the analytic results derived by Wu [6]. Our results indicate that in the presence of zero-mean gaussian random manufacturing errors the expected positioning accuracy of an arbitrary robot will be nonzero. While a reduction in manufacturing errors will tend to increase robot positioning accuracy, this approach by itself will not necessarily eliminate the expected positioning errors inherent to robots which utilize a design model control strategy.

4.2 Machining and Assembly Errors

To evaluate the effect of positional and orientational machining and assembly errors upon robot performance, five more simulations were performed. In these simulations, the standard deviations of the positional and orientational errors, σ_p and σ_o were varied simultaneously from 0.01 cm and 0.01 deg to 0.05 cm and 0.05 deg, respectively, while the encoder calibration error standard deviation, σ_e, was fixed to 0.1 deg. The standard deviations of the 34 performance indices are again plotted in [4,5] as a function of σ_p and σ_o. These sample standard deviations are based upon the performance of 500 robots.

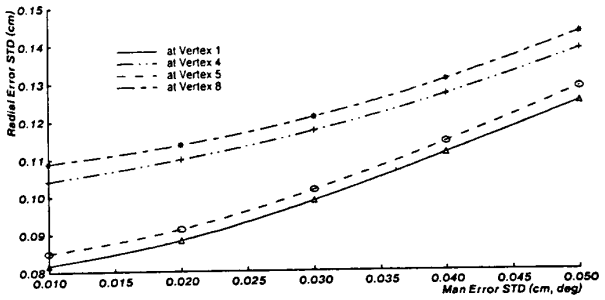

Figure 4: Radial Position Error Standard Deviation as a Function of σ_p and σ_o.

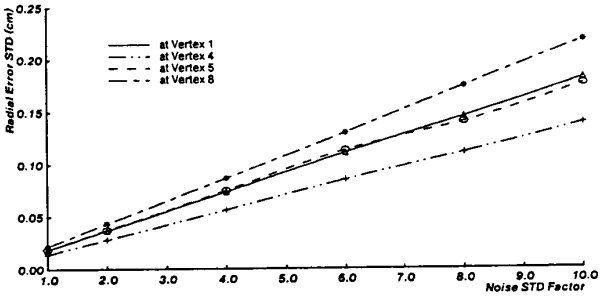

Figure 5: Radial Position Error Standard Deviation as a Function of the Measurement Noise Factor k.

The variations in the performance index standard deviations in these simulations are quite different from those in Section 4.1. For instance, consider Figure 4 where the radial position error standard deviations at Vertices 1, 4, 5, and 8 are plotted as a function of σ_p and σ_o. In contrast to Figure 3, the relationships shown in Figure 4 are distinctly nonlinear. Similar nonlinear relationships are exhibited by the remaining performance index standard deviations. Except for the Vertex 1 and Vertex 4 absolute orientational error standard deviation, the performance index standard deviations all increase monotonically as σ_p and σ_o increase. Furthermore, the slopes of all 34 curves increase monotonically.

In further contrast to the curves in Figure 3, the curves in Figure 4 do not all diverge from one another. For example, as σ_p and σ_o increase the difference between the variability in the radial positioning error at Vertex 1 and at Vertex 5 decreases from 0.0224 to 0.0138. In some cases the curves may even intersect.

The variations in the performance index means as a function of σ_p and σ_o are much more predictable than when as a function of σ_e. As σ_p and σ_o increase, all of the performance index means increase monotonically. For D-Model Control, the presence of positional and orientational manufacturing errors leads to an average decrease in end-effector positioning accuracy.

The effect of positional and orientational manufacturing errors on the variability of end-effector positioning accuracy is more complex than that for encoder calibration errors as evidenced by the nonlinear relationships in Figure 6. However, the effect of these manufacturing errors upon the expected end-effector positioning accuracy of a robot is much more simple than for the effect due to encoder calibration errors. Finally, the sensitivity of the performance index standard deviations to simultaneous variations in σ_p and σ_o is less than the sensitivity of the performance index standard deviations to variations in σ_e.

4.3 Sensor Measurement Errors

In [4], equations were derived empirically to describe the relationship between feature estimate accuracy and sensor system accuracy, σ, as well as the identification parameters, N_i and R_i. Unfortunately, the effect of feature estimate accuracy upon the performance of an S-Model Controller cannot be derived analytically. Therefore, the simulator was applied to empirically determine the qualitative relationship between measurement noise standard deviation, σ, and the expected performance of a Puma 560 robot which uses signature-based control. Six separate simulations were conducted each using a different value of σ in (11), to identify the robot's arm signature. In each simulation, the value of (sigma) was computed according to (11) and then multiplied by a factor k ranging from 1.0 to 10.0.

The standard deviations of the 34 performance indices as function of k are plotted in [4,5]. The plot of the Radial Position Error standard deviation is reproduced below in Figure 5. The standard deviations of all the performance indices increase linearly with the noise measurement factor. The more inaccurate the sensor system is the more inaccurate the identified signatures are and hence the greater the variability in the performance of the S-Model Controller. The significance of these results is that the sensitivity of the performance of the S-Model Controller to variations in sensor system accuracy is consistent with the relationships in [4] which govern the feature estimate accuracy. In these equations, the orientational accuracy of the estimated planes-of-rotation and positional accuracy of the estimated centers-of-rotation are both directly proportional to the measurement noise standard deviation. Clearly, feature estimate accuracy has a direct impact upon the performance of an S-Model Controller.

Nineteen of the performance indices are defined such that they can take on any real value (e.g., the various Line Length Error indices). Using S-Model control the mean values of these nineteen indices are all zero which is in strong contrast to the effect which manufacturing errors have upon the performance of the D-Model Controller. The significance of this is that on the average we can expect the S-Model identification algorithm to correctly identify the true kinematic parameters of a robot and thus eliminate end-effector positioning errors. For the remaining 15 performance indices, the index's mean values are directly proportional to the measurement noise standard deviation, σ.

4.4 Number of Measurements

The effect of feature estimate accuracy upon the performance of an S-Model Controller can not be derived analytically. Therefore, the simulator was again applied to empirically determine the qualitative relationship between N_i and the expected performance of a Puma 560 robot which uses S-Model Control. Five separate simulations were conducted each using a different number of measurements per joint, N_i, to identify the robot's arm signature.

The standard deviations of the 34 performance indices for N_i = 20, 40, 60, 80, and 100, are plotted in [4,5]. For comparison purposes, the plot of the Radial Position Error standard deviation is reproduced below in Figure 6.

In general, the standard deviations of all 34 performance indices are inversely proportional to the square root of N_i. By increasing the number of measurements used in identifying a robot's arm signature, substantial increases in end-effector positioning accuracy can be achieved. For instance, in Figure 6 the Radial positioning errors are reduced by a factor of 2.34. Such findings are consistent with the analytic relationships developed in [4] regarding feature estimate accuracy. In these relationships, the orientational accuracy of the estimated planes-of-rotation and the positional accuracy of the estimated centers-of-rotation are both inversely proportional to the square root of the number of measurements. Feature estimate accuracy thus has a

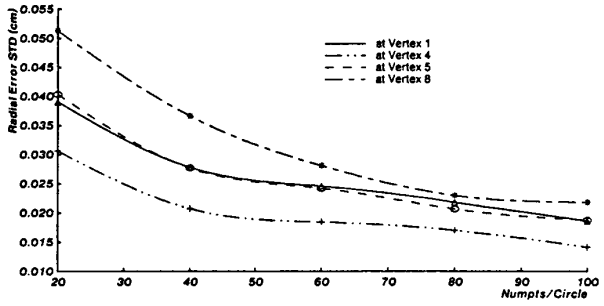

Figure 6: Radial Position Error Standard Deviation as a Function of the Number of Points per Circle, N_i.

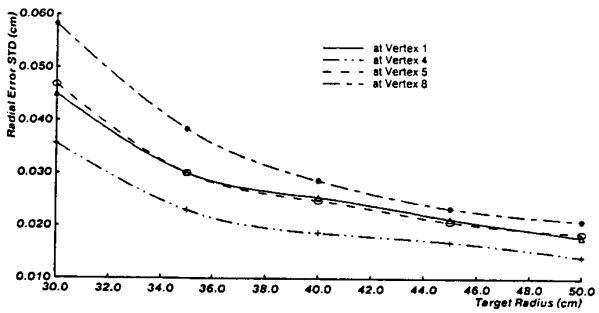

Figure 7: Radial Position Error Standard Deviation as a Function of Target Radius.

direct impact upon the performance of the S-Model Controller.

Consider again the nineteen performance indices which are defined such that they can take on any real value. Using signature-based control the mean values of these nineteen indices are all zero (i.e., negligible in comparison with the corresponding standard deviations) which is in strong contrast to the effect which manufacturing errors have upon the performance of D-Model Control. On the average we can expect the S-Model identification algorithm to correctly identify the true kinematic parameters of a robot and thus eliminate end-effector positioning errors. For the remaining 15 performance indices, the index's mean values are inversely proportional to the square root of N_i.

4.5 Effect of Target Radius

In this section, we apply the simulator to analyze the relationship between end-effector positioning accuracy and the target radii. Naturally, the values of certain target radii will have greater impact upon the overall performance of the robot than will others. Since it would be impractical to simulate all the possible combinations of the values of the six target radii and interpret the results, we limited our analysis to the situation in which all the target radii are equal. Thus in each simulation, $R_i = R_2 = \ldots = R_6 = R_{nom}$. Five simulations were performed in which the value of R_{nom} is varied between 30.0 cm and 50.0 cm. The standard deviations of the 34 performance indices as a function of R_{nom} are plotted in [4,5]. The plot of the Radial Error standard deviations is reporduced in Figure 7.

While the parameter N_i has an effect upon both the accuracy of the identified planes-of-rotation and centers-of-rotation, the parameter R_{nom} only effects the accuracy of the planes-of-rotation. Taken independently, N_i and R_{nom} are both inversely proportional to the accuracy of the identified planes-of-rotation. From previous findings we might thus expect that increasing R_{nom} would have less of an overall effect upon the performance index standard deviations than does N_i. However, from Figure 7 and the remaining plots this is not the case. The curves in Figure 7 are approximated more closely by an inverse square relationship rather than by an inverse relationship. The cumulative accuracy of an identified arm signature appears to be more sensitive to orientational errors than positional errors. This could be explained by the fact that orientational errors, especially those involved in the description of a manipulators positional subsystem, not only propagate into orientational errors at the end-effector but they are also transformed and amplified into positional errors at the end-effector. The amount of amplification being proportional to the lengths of the links and the configuration of the robot.

The variations in R_{nom} have no effect upon the mean values of the standard deviations of the performance indices which are defined over all the real numbers. Again, the mean values for these 19 performance indices are zero. For the remaining 15 performance indices, the indices' mean values are, in general, inversely proportional to R^2_{nom} .

These findings suggest that the best strategy for increasing signature accuracy is to first increase the target radii followed by increases in the number of measurements. The cost of increasing the number of measurements is the increased time required to obtain all the target measurements. This cost is incurred during the identification of each individual signature. In contrast, the cost associated with increasing R_{nom} is simply the reconstruction of a set of target mounting fixtures. This cost has only to be incurred once since a single set of fixtures can be applied to all robots of the same model.

5 Conclusions

In this paper, we have formualted a methodology by which to evaluate and compare the statistical performance of a conventional design model-based kinematic controller and a signature-based kinematic controller. The design model controller incorporates the robot kinematic parameters obtained from the mechanical design specifications. In contrast, the signature-based controller incorporates the robot kinematic parameters estimated by the S-Model Identification algorithm. Furthermore, we have applied this methodology to evaluate and compare the statistical performance of these two controllers when applied to control the joint positions of a Puma 560 robot. This work complements our experimental evaluation of the performance of seven Puma 560 robots described in [3,4,5].

Our methodology utilizes Monte-Carlo simulation techniques for the generation of end-effector positioning and orienting error distributions. The evaluation of the performance of the D-Model Controller, requires the formulation of a robot kinematic model which explicitly incorporates the manufacturing errors as parameters. A kinematic model of a Puma 560 robot with manufacturing errors was developed and serves to illustrate how such a model could be developed for other robots. The performance of the S-Model Controller depends upon the types of the errors introduced into the arm signature identification process and not upon the presence of robot manufacturing errors. Having developed a model of our ultrasonic sensor system, we simulated the arm signature identification process using a Puma 560 robot and applied the identified signatures to control the robot in performing a series of standardized tasks. The positioning and orienting accuracy of the end-effector was then used to evaluate how well the S-Model Controller performed. This approach can be applied to analyze the statistical performance of the S-Model Controller for any robot with any type of sensor system by interchanging simulator modules.

In Sections 4.1 and 4.2, we applied our Monte-Carlo simulator to investigate the relationship between encoder calibration errors and robot performance, and the relationship between machining and assembly errors and robot performance, respectively. From this investigation we have discovered that

- The performance indices standard deviations increase linearly with the standard deviation of the encoder calibration errors.
- The performance indices standard deviations increase nonlinearly and monotonically with simultaneous changes in the standard deviations of the positional and orientational manufacturing errors.
- The performance of a D-Model Controller is more sensitive to encoder calibration errors than to the positional and orientational manufacturing errors.
- The expected performance of a D-Model Controller is a function of both the means and standard deviations of the manufacturing errors.

In practice then, the average performance of a robot using design model based control will be less than perfect. This finding clearly supports our earlier claim that arm signature identification techniques are needed to improve end-effector positioning and orienting accuracy.

In Sections 4.3 - 4.5, we have also applied our Monte-Carlo simulator to understand the relationships which govern the performance of an S-Model Controller. Specifically, we were interested in the relationships between the sensor system accuracy, the identifier parameters N_i and R_i, and the end-effector positioning accuracy. We have found that,

- The performance indices' standard deviations are directly proportional to the sensor measurement error standard deviation σ and that this relationship is consistent with the analytic relationships between σ and the measures of feature estimate accuracy derived in [4].
- The performance indices' standard deviations are inversely related to the square root of the number of measurements, N_i and that this relationship is consistent with the analytic relationships between N_i and the measures of feature estimate accuracy derived in [4].
- The performance indices' standard deviations are inversely related to the square of the target radii.
- The performance of an S-Model Controller is more sensitive to orientational identification errors than to positional identification errors.
- The expected performance of an S-Model Controller depends only upon the expected value of the measurement errors.

The identifier parameters, N_i and R_i, provide a simple mechanism for increasing the accuracy of the identified arm signatures and hence the performance of an S-Model Controller by a predetermined amount. These parameters can thus be used to reduce the need for extremely accurate sensor systems to measure the targets positions. Finally, the S-Model identification algorithm and control approach will, on the average, provide perfect end-effector positioning and orienting accuracy in the presence of zero-mean target measurement errors. The elimination of biases in the expected performance of a robot is an important advantage of the S-Model identification algorithm and control approach.

REFERENCES

[1] GP-8-3D Sonic Digitizer Operator's Manual Science Accessories Corporation, Southport, CN, 1985.

[2] Stone, H.W., Sanderson, A.C. and Neuman, C.P. Arm Signature Identification. In the Proceedings of the IEEE International Conference on Robotics and Automation, pages 41-48. April, 1986. San Francisco.

[3] Stone, H.W. and Sanderson, A.C. A Prototype Arm Signature Identifi cation System, In the Proceedings of the IEEE International Conference on Robotics and Automation, pages 175-182. April, 1987. Raleigh, NC.

[4] Stone, H.W. Kinematic Modeling, Identification, and Control of Rob otic Manipulators. PhD thesis, Carnegie-Mellon University, Pittsburgh, PA, December 1986.

[5] Stone, H.W. Kinematic Modeling, Identification, and Control of Rob otic Manipulators. Kluwer Academic Publishers, Boston, 1987.

[6] Wu, C. The kinematic Error Model for the Design of Robot Manipulat ors. In the Proceedings of the American Control Conference, pages 497-502. San Francisco, CA, June, 1983.