LA-UR-02-3576

Approved for public release; distribution is unlimited.

Title: METAMODELS FOR PLANAR 3R WORKSPACE OPTIMIZATION

Author(s): Cameron J. Turner, NMT-15
Los Alamos National Laboratory

Submitted to: Proceedings of DETC'02:
ASME 2002 Design Engineering Technical Conferences & Computers and Information in Engineering Conference

Montreal, Canada, September 29-October 2, 2002

Los Alamos
National Laboratory

Proceedings of DETC'02:
ASME 2002 Design Engineering Technical Conferences and Computers and Information in Engineering Conference
Montreal, Canada, September 29-October 2, 2002

DETC2002/CIE-34500

METAMODELS FOR PLANAR 3R WORKSPACE OPTIMIZATION

Cameron J. Turner
Department of Mechanical Engineering
The University of Texas at Austin

ABSTRACT

Robotic workspace optimization is a central element of robot system design. To formulate the optimization problem, the complex relationships between design variables, tuning parameters, and performance indices need to be accurately and efficiently represented. The nature of the relationships suggests that metamodels, or models of the models, should be used to derive suitable objective functions. A comparison of two metamodeling techniques for robotic workspace optimization problems for several trial cases suggests that non-uniform rational B-spline models, derived from computer graphics and computer-aided design techniques, are as suitable as response surface models to solve planar 3R workspace optimization problems. Promising nonlinear modeling results with B-spline models suggest future work is justified and performance gains can be realized.

INTRODUCTION

During the process of design, engineers are ultimately confronted with the challenge of determining how to modify design variables in order to achieve or exceed desired performance criteria. For simple systems, or for experienced engineers, an Edisonian, or experiential approach to design is often used in lieu of a formal design methodology. However, this approach may fail if the engineer lacks sufficient experience to be able to make appropriate decisions, or if the complexity of the system exceeds the experience of the engineer. This limitation is common for many design problems of current interest where high performance and limited design experience exists. In such problems, the number of performance criteria, typically highly non-linear functions (perhaps unknown in closed form) of a large number of design variables, that must be considered simultaneously far exceeds the number that a human engineer can comfortably contemplate.

While the complexity of many current design problems surpasses the experience and capacity of many engineers, the development of complimentary computational capabilities that can be applied to the benefit of the designer seems to be ever more feasible. Very sophisticated, commercially available software systems are used to predict the performance of proposed designs. Computers are able to process far more data, at a far greater speed than any engineer can, but adequate tools to combine the predictions of multiple performance simulations for informed decisions making are unavailable or problematic. This data, often from disparate sources, must be combined into a single coordinated representation for effective analysis and visualization. This is accomplished through a family of techniques known as metamodeling. Metamodeling uses experiments and models as the basis for a higher-level model of models, known as a metamodel.

The most popular metamodeling technique, Response Surface Models (RSM) uses quadratic approximations [14] and is difficult to implement for more than 10 variables. [13,26] Real problems are often inadequately modeled with quadratic representations and have many more than 10 variables (dozens or hundreds are common). For complex, nonlinear problems, these limitations can lead to ineffective metamodels.

Metamodels based on spline theory, B-Spline Models (BSM) may be able to overcome these limitations. The focus of this work is a preliminary study of the performance of BSMs compared to RSMs. For this study, a robotic workspace design problem was selected. This design problem can be formulated with a limited number of design variables, making it computational tractable for a preliminary study, and yet this system can exhibit highly nonlinear behaviors.

Robotic system design is inherently concerned with the definition of robot workspaces. All robotic workspaces are subsets of the Reachable Workspace of a robot, or the set of points that the robot can reach with its tooling. An appropriately designed robot will ensure that the task manifold is contained within the reachable workspace [18]. General-purpose robot designs maximize the reachable workspace with respect to geometric constraints that describe the robot's operating region while maintaining a dexterous workspace [29]. This is the basis for workspace optimization.

While most robots are spatial mechanisms, planar 3R mechanisms, such as that shown in Figure 1, (comprised of three rotary joints operating in a plane) are an important group of mechanisms that can be used to represent the limbs of many biological and mechanical mechanisms [8]. Despite their significance, previous research has addressed few of the kinematic serial planar manipulator workspace optimization issues [3].

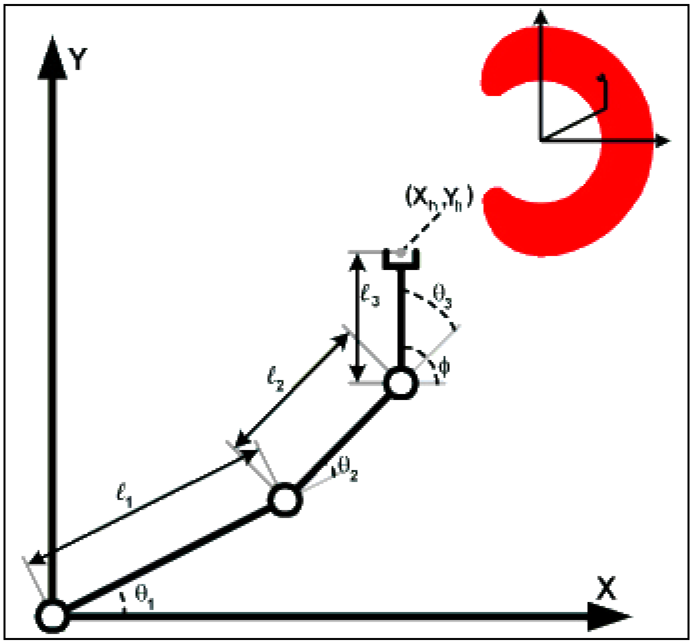

Figure 1. *3R Serial Planar Configuration and reachable workspace (in red) used to compare RSM and BSM performance.*

Like many complex design problems, analytical tools supporting workspace optimization are limited [20]. Due to the introduction of trigonometric functions to describe the position of the rotary joints, the resulting systems models exhibit nonlinear behaviors. Therefore, the use of a metamodel, or a simplified model of the original model can be used to identify optimal solutions for complex problems [30]. Metamodels are commonly used in nonlinear, complex, and multidisciplinary optimization problems. A simplified computationally beneficial metamodel is derived for the optimization problem objective function.

Two metamodeling approaches are compared in this paper. Response Surface Models (RSMs) are based on polynomials derived from Design of Experiments Methodologies. A second approach uses Non-Uniform Rational BSplines (NURBs), a general piecewise polynomial representation derived from spline theory and employed in computer graphics, to produce a B-Spline Model (BSM).

The remainder of this paper reviews metamodeling literature and develops a set of metamodel criteria to compare the suitability of these two techniques as solutions to planar 3R workspace optimization problems. Using successively more complex formulations and metamodel selection criteria, metamodel performance estimates support the conclusion that BSMs are promising and at least as suitable as RSMs to solve planar 3R workspace optimization problems. Supporting software, and further work is needed.

1. METAMODELING

Metamodels are generally employed in three cases. The first case is when accurate analytical models are not available and a metamodel must be derived from experimental data. The second case is when analytical models are available, but require more computational effort to use than is feasible. The third case combines multiple analytical and/or experimental models into a single metamodel often for multidisciplinary optimization (MDO) efforts. The relationship between the actual system, direct models and metamodels is shown in Figure 2.

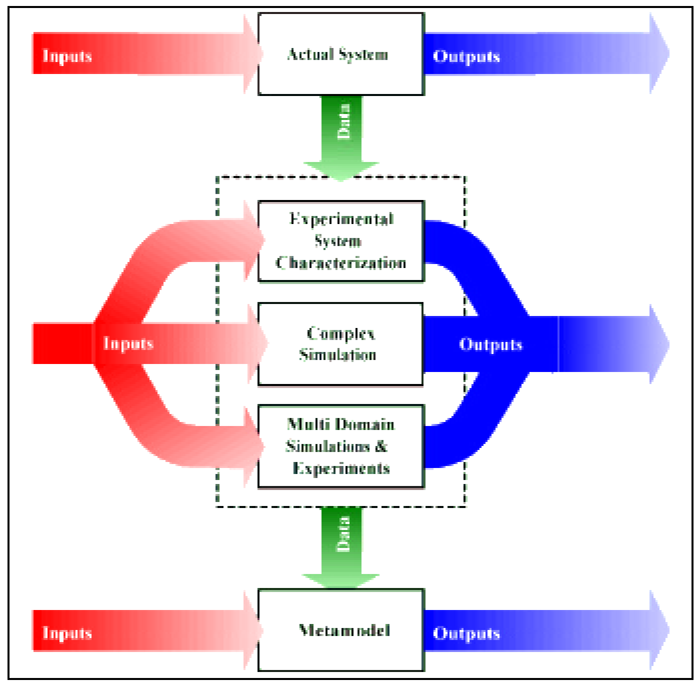

Figure 2. System, Model and Metamodel Relationships.

The most common form of a metamodel uses Design of Experiments techniques to formulate an RSM, which is an implicit polynomial representation of the relationship between design variables and performance indices [27]. This approach can be contrasted with the BSM parametric polynomial form.

1.1. RESPONSE SURFACE MODELS

RSMs use an implicit polynomial formulation, denoted as P(x,y) in R^2 space, where x and y are design variables, and P is a performance index. The polynomials used in RSMs typically are either linear or quadratic polynomials [26,27,30]. Equation 1 shows the form of a single variable RSM, while Equation 2 shows the form of a dual variable RSM [27].

$$P(x) = \sum_{i=0}^{n} \beta_i x^i \quad (1)$$

$$P(x, y) = \sum_{i=0}^{n} \sum_{j=0}^{n} \beta_{j+n \cdot i} x^i y^j \quad (2)$$

Where β represents the coefficients of each term, and n defines the order of the polynomials used to define the RSM [27]. RSMs can be used to model curves in R^2 space as well as surfaces in $R^{n>2}$ space. These formulations may be extended to cases with additional variables, although ten variables are generally considered the practical limit [13,26]. RSMs are usually derived as linear models, which are replaced with quadratic models at the end of the application for model confirmation [14]. This RSM formulation is generalized in Equation 3.

$$P(D_V) = \{\beta\}\{N_P(D_V)\}^T \quad (3)$$

Where $\{N_p(D_v)\}$ is a vector of the terms resulting from a power series expansion of the design variables, D_v, that serves as the basis for the RSM. The RSM is "fit" to the data using single or multivariate least-squares regression techniques [4]. As a result, RSMs can provide predictive capabilities [14]. The RSM properties are determined by its implicit form and by the techniques used to "fit" the model to the available data, so implicit form properties are discussed next.

1.2. IMPLICIT REPRESENTATIONS

Implicit geometric representations are well known, and are often the first surface representation introduced to students. The point set satisfying Equation 4 defines an implicit curve, while the point set satisfying Equation 5 defines an implicit surface [5].

$$f(x, y) = 0 \quad (4)$$

$$f(x, y, z) = 0 \quad (5)$$

Implicit forms have certain advantages. They are unique formulations with respect to a multiplicative constant [22]. Implicit representations readily represent unbounded curves and surfaces, and provide easy determination of the membership of a point in the point set defining an object [22]. Implicit surfaces can be readily extended to R^n spaces by increasing the dimensionality of the independent variables. From the perspective of an analyst, the coefficients of an RSM provide considerable information about the relative significance of D_v's within the RSM.

However, implicit forms also have disadvantages. They are dependent upon the choice of coordinate system, which may lead to numerical instabilities in their evaluation and difficulties in their use, affecting their ability to match segmented RSMs at their boundaries [24]. Furthermore, implicit objects are more difficult to evaluate at equal intervals over the representation, an important step for meshing and data visualization [24]. Implicit formulations include planar and spatial curves, as well as hyperdimensional surfaces.

The polynomial representations produced from implicit RSMs have been successfully used as the basis for defining the objective function for multiple optimization algorithms [14], including gradient [19], simulated annealing [31], and genetic algorithm type methods [32]. Derivatives of polynomials are easily calculated for gradient methods.

Difficulties with implicit representations have led to a significant body of work surrounding parametric representations for computer graphics and computer-aided design.

1.3. PARAMETRIC REPRESENTATIONS

Parametric representations use explicit relationships between the coordinate system and one or more independent parameters. A parametric curve is described with a single independent parameter, while a parametric surface utilizes two independent parameters. Higher order parametric objects can be constructed with additional parameters. The resulting object is described by a vector valued function such as P(u,v) shown in Equation 6 for a surface in R^3 space [24].

$$f(x(u,v), y(u,v), z(u,v)) = P(u,v) \quad \text{for } a \le u, v \le b \quad (6)$$

Compared with implicit representations, parametric formulations have advantages. Low order parametric formulations are readily extensible to higher order spaces, and higher order object representations can be derived. The parameterization of these objects makes the form of the object coordinate system independent [22]. Parameter bounds make these representations ideal for the modeling and segment blending [22]. The role of the parameter makes these forms more useful for interpolation than for prediction [14]. Parametric forms are amenable to generating uniform meshes, enabling easy computer representations [22].

Unlike implicit forms, parametric forms do not have unique solutions [22]. Parametric objects generally require the storage of more information than an implicit representation's polynomial coefficients, and the coefficients of a parametric representation do not necessarily convey the same information about the relative importance of individual terms, as is the case with implicit formulations. Parametric formulations also can generate incorrect features, such as loops and folds in the representation [22].

It is often possible to convert implicit representations into parametric forms, and vice versa [16]. Before rendering, implicit representations are often converted to a generalized type of parametric representation, known as a Non Uniform Rational B-Splines or NURBs. NURBs are the de facto standard for geometric descriptions in computer graphics and computer-aided design applications [22].

A polynomial can be represented as a NURB, and so, given a NURB, an appropriate polynomial expression can be derived. NURB derivatives are also explicitly defined [22,23,24]. Therefore, as equivalent polynomial representations, NURBs also can be used as the basis for many optimization algorithms.

1.4. B-SPLINE MODELS

The BSMs developed for this work are based on NURBs, which are defined for a curve by Equation 7 [24].

$$P(u) = \frac{\sum_{i=1}^{n_C} B_i w_i N_{i,k}(u)}{\sum_{i=1}^{n_C} w_i N_{i,k}(u)} \quad \text{for } a \le u \le b \tag{7}$$

Where {B} is a vector defining the location of the defining n_C control points in R^{n+1} space, w defines the weight of a particular control point, and $N_{i,k}(u)$ is the B-spline basis function in terms of the parameter u, as defined by Equations 8 and 9 [24].

$$N_{i,k}(u) = \left(\frac{u - x_i}{x_{i+k-1} - x_i}\right) N_{i,k-1}(u) + \left(\frac{x_{i+k} - u}{x_{i+k} - x_{i+1}}\right) N_{i+1,k-1}(u) \tag{8}$$

$$N_{i,1}(u) = \begin{cases} 1 & if \;\; x_i \le u < x_{i+1} \\ 0 & otherwise \end{cases} \tag{9}$$

Where {x} is the knot vector, a sequence of values defining the region of control point influence within the NURB. Equations 8 and 9 are subject to the conditions given by Equation 10 [24].

$$\begin{gathered} 0/0 = 0 \\ \sum_{i=1}^{n+1} N_{i,k}(u) \equiv 1 \;\; \forall k \text{ and } u, \text{ and} \\ if\; x_{i+1} = 1,\; N_{i,1}(u) = \begin{cases} 1 & if \;\; x_i \le u \le x_{i+1} \\ 0 & otherwise \end{cases} \end{gathered} \tag{10}$$

Higher order objects, such as surfaces, are produced with a tensor product, resulting in a grid of control points, and the multiplication of Equation 7 by additional basis functions associated with the additional parameters. Control points, associated weights, and one or more parameters and knot vectors define a NURB. Each of these parameters lends additional flexibility to NURBs, producing a universal curve definition [10].

The BSMs used for this comparison have all weights set to 1, a typical approach [22,24], and utilize spline-fitting techniques developed by Legault [17] and Turner [28] that iteratively reduce the maximum error in a spline model. This approach to fitting BSMs to data is distinctly different from traditional BSM metamodel approaches that match control points to data points in a one-to-one relationship [27].

The increased complexity of BSMs is a disadvantage, requiring the storage of control points locations and knot vectors, and because the additional flexibility of NURBs results in an under constrained system of equations that can result in poor results [21]. A literature review found few references on BSM representations, apparently due to the difficulties in BSM implementation [1,2,27]. Most work was conducted in the early to mid 1990s and no references specific to BSMs using NURBs were found.

Potential BSM advantages are recognized. Practical studies have indicated that in complex, nonlinear, multidisciplinary design environment, additional metamodel flexibility is beneficial [25]. In particular, NURBs have been found to be numerically robust, capable of accurately representing many geometric forms, and have a broad set of supporting algorithms [21]. Like a polynomial, the BSM order determines its computational complexity. BSMs also benefit from the use of specialized graphics acceleration hardware for rapid NURBs computation [10]. BSMs also allow for local data changes without requiring global model recalculations [22].

1.5. METAMODELING CRITERIA

Three features distinguish RSMs and BSMs. BSMs use low order polynomials valid over small regions of a global data set, while the RSM polynomials are global. BSMs use B-spline basis functions, while RSMs use power basis functions. Finally, BSMs define their shape with control points while RSMs use their polynomial coefficients.

In order to compare the suitability of these two metamodeling techniques, a set of criteria are needed. Optimization algorithms did not provide suitable criteria, as both techniques can be used by a variety of optimization algorithms. The central issue between RSMs and BSMs in optimization is the model accuracy and computational complexity. In essence, how good is the model?

Hussein, et al, [12] recently proposed a set of eight criteria to select appropriate metamodels. Two of these criteria are concerned with data acquisition issues, which is beyond the scope of this work. The remaining six criteria: 1) Computational Complexity, 2) Model Accuracy, 3) Model Visualization, 4) Model Flexibility, 5) Stability with respect to New Data, and 6) Commercial Software Availability were adopted to compare RSMs and BSMs [12].

The first two criteria are concerned with the model's computational complexity and accuracy. A model that does not produce accurate results, or that requires excessive resources to calculate is a poor metamodel [12,25]. The third criterion is concerned with the subsequent ability to visualize the model and thus, provide insight to the analyst.

Criterion 4, model flexibility is important for optimization. The chief challenge when using metamodels for optimization is achieving a suitably accurate and smooth representation of an arbitrary data set [12].

The fifth criterion is considered when the metamodel is provided with new data and must incorporate this new information [12]. The sixth criterion is commercial software availability. While this criterion is biased towards popular, traditional and existing techniques, it is a reasonable basis to judge metamodels in a non-research setting [12].

2. ROBOTIC WORKSPACE OPTIMIZATION

Little work has been conducted on 3R serial planar mechanism kinematic optimization [3], and most focuses on dynamic properties. A few kinematic criteria have been considered including reachable workspace volume (RWVC), Jacobian condition number (JCNC), and joint range availability (JRAC) [6,8,29].

Of these criteria, both RWVC and JRAC have closed form expressions. JRAC and the JCNC are local criteria, and thus are functions of the manipulator joint angles, while RWVC is a global property of the manipulator. Since RWVC is a fundamental robot design element, it is used extensively in section 4. JCNC was also selected due to more interesting criterion properties. Both criteria are functions of the link lengths.

JCNC is a particularly valuable and challenging criterion, derived from the Jacobian matrix of a manipulator [11]. JCNC measures the proximity of a manipulator to singularities [15]. As a robot approaches a singularity, its controllability is reduced [15], the manipulator's ability to exert or react forces is altered [33], and the dexterity of the robot is limited [11]. Further, JCNC is related to JRAC [15]. Several JCNC forms have been proposed and used [15]. A condition number based on the infinity norm of the Jacobian matrix is used in section 4.

Metamodel results used to represent RWVC and JCNC for the robotic system described in section 3 are shown in section 4. Closed form expressions for JCNC gradients are not available, complicating optimization with JCNC and requiring metamodels [15].

2.1. PROBLEM DESCRIPTION

The 3R serial planar robot configuration used for this work was previously shown in Figure 1. The position of the end effector, as a function of the joint angles can be derived from the system geometry. This model can be differentiated with respect to the joint angles to define the Jacobian matrix, a standard element in the formulation of most robotic system models. The system model is subject to the assumptions described by Equations 11 and 12.

$$|\theta_i| \le \pi/2 \quad for\ i=1,2,3 \tag{11}$$

$$and\ \ell_3 = 0.1 \tag{12}$$

Equation 11 reduces the effect of singularities on the system. Two singular cases remain, when θ_2=0, and a boundary singularity condition due to the reachable workspace boundary. Both cases are denoted by the inability of the manipulator to simultaneously specify x, y, and ϕ [7]. Equation 12 also was applied to set minimum link lengths for links 1 and 2.

The metamodel goal is to reveal and define relationships between design variables, D_v, tuning parameters, T_p, and performance indices, P_i. These variables comprise the system design space shown in Figure 3. Metamodels are used to define functional relationships in the form of Equation 13.

$$P_i = f(\{D_V\},\{T_P\}) \tag{13}$$

Based on the system shown in Figure 1, and the assumptions made in section 3.1, the system design variables, tuning parameters, and performance indices are defined in Table 1. Additional terms such as those for dynamic characteristics also exist, but were not considered. The variables used to compare metamodels are shown in Table 1 in red.

Table 1. *3R Serial Planar Manipulator Design Space Axes.*

D_v:	L_1, l_2, l_3, β, x_{base}, y_{base}, τ_1, τ_2, τ_3, …
T_p:	θ_1, θ_2, θ_3, ω_1, ω_2, ω_3, α_1, α_2, α_3, …
P_i:	**RWVC, JCNC**, JRAC, Payload, …

These variables define five trial cases, two of which are P_i's that are functions of a single active variable (2D cases), and three of which are P_i's that are functions of two active variables (3D cases). The remaining terms were artificially held constant. Sections 3.2 and 3.3 describe these cases.

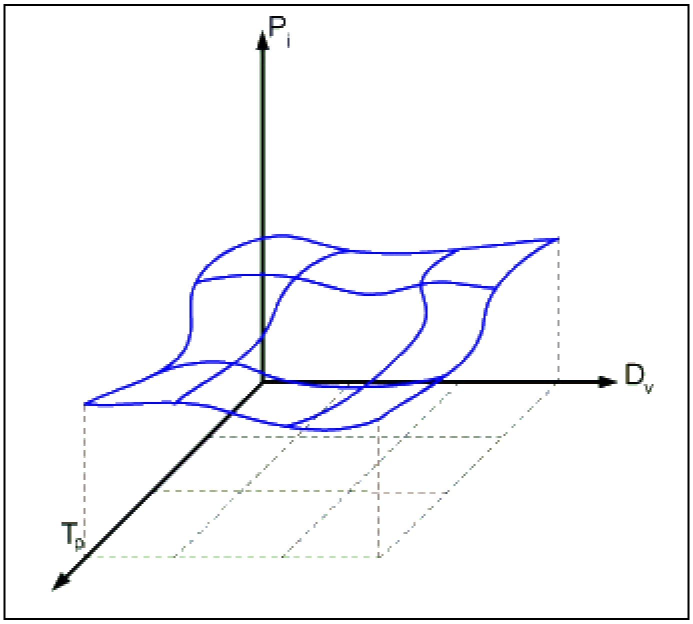

Figure 3. *The Design Space.*

2.2. 2D FORMULATIONS

Two cases were examined where a performance index was written as a function of a single design variable, resulting in a 2D metamodel. The first case defines the lengths of link 1 and 2 as a function of a single parameter, β, according to Equations 14 and 15. The objective is to maximize the reachable workspace for a unit manipulator length. This can be considered to be a normalized, nondimensional length design problem.

$$\ell_1 = 0.1 + 0.7\beta \tag{14}$$

$$\ell_2 = 0.1 + 0.7(1-\beta) \tag{15}$$

The performance index of interest is the RWVC, a global criterion that is a function of the link lengths, or in this case, a function of solely β.

The second case considers the JCNC for a fixed set of link lengths (0.5,0.4,0.1). To simplify this problem to 2D, the link lengths were fixed, and the tool orientation was set at $\phi=30^o$ by fixing $\theta_2=30^o$, and $\theta_1=-\theta_3$. As a result JCNC is a function of only θ_1.

As a local criterion, the objective of JCNC is to minimize the JCNC value throughout the reachable workspace. To be compatible with RWVC, the P_i is defined as the negative sum of the 2-norm over the reachable workspace space volume, shown in Equation 16.

$$P_i = -\sqrt{\int_V JCNC(\theta_1)dV} \quad (16)$$

2.3. 3D FORMULATIONS

Three cases were examined where the P_i is a function of two active variables. The first case is an extension of the initial 2D case where the requirement for a manipulator of unit length has been removed. This is a dimensioned manipulator design problem. Thus, the RWVC is now a function of the link lengths 1 and 2.

A more complicated case occurs if the size of a manipulator is constrained by a geometric boundary. While the initial 2D case addresses the design problem associated with a no collision condition, allowing for collisions with the boundary may enhance the overall manipulator workspace volume. This case calculates the ratio of the reachable workspace to the potential workspace volume within a 2-unit square centered on the robot base.

The third 3D case considered is for the JCNC where the link lengths remained fixed at (0.5,0.4,0.1) and $\theta_2=30^o$. Without a condition on, ϕ, JCNC becomes a function of θ_1 and θ_3.

2.4. IMPLEMENTATION

Data for these trials was obtained via simulations created within the program MathCad™ for a 900MHz PC computer. All subsequent calculations to derive and evaluate RSMs and BSMs were performed within this same environment. Algorithms defining RSMs available within MathCad™ were used for RSM generation, while algorithms defined by Legault [17] and Turner [28] where simulated within this environment. No dedicated supporting software algorithms were created due to time constraints imposed upon this research. The preliminary use of these algorithms via MathCad™ allowed for preliminary results to be obtained without the development of optimized supporting programs. This limited the size of the problems that could be solved to levels that are far less than those solved with dedicated software by both Legault [17] and Turner [28].

3. MODEL COMPARISONS

Using the five cases presented in sections 3.2 and 3.3, and the criteria from section 2.5, RSM and BSM metamodels for each case were derived and their accuracy was compared. Since RSMs are typically limited to quadratic models [14], quadratic RSMs are used as the basis for comparison with quadratic BSMs. Thus the metamodels generally have equivalent computational complexities.

3.1. 2D RSM VERSUS SPLINE MODELS

The first 2D case fits RSM and BSM metamodels to data representing the RWVC versus the link ratio, β. The data is well fit by both the RSM and BSM metamodels, with comparable errors, and correlations exceeding 99% as calculated from Pearson's correlation coefficient [4]. Figure 4 shows the model results. The 2-segment BSM accuracy in this trial is equivalent to the quadratic RSM used.

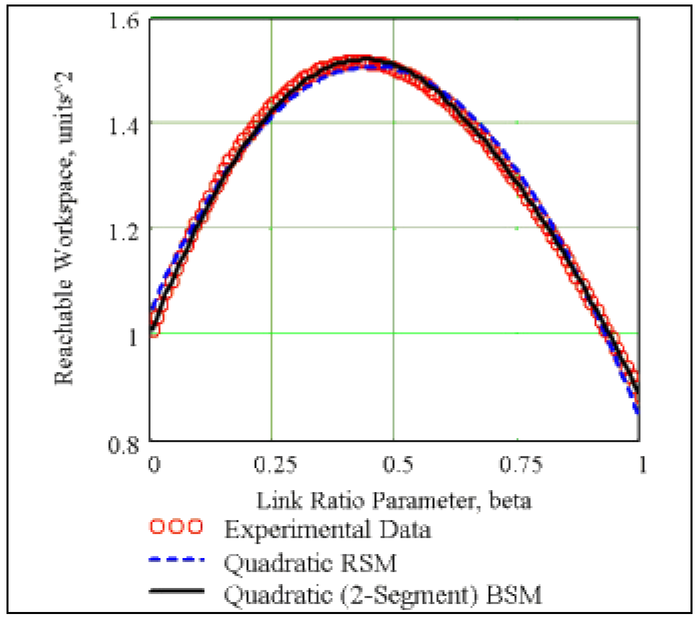

Figure 4. *Case 1: Metamodel Accuracy.*

The case 2 data is considerably more complex and nonlinear. A quadratic RSM results in a poor fit, with large errors and a correlation of <50%. A 5-segment quadratic BSM produces a model with smaller errors and a correlation of more than 90%. Results are shown in Figure 5.

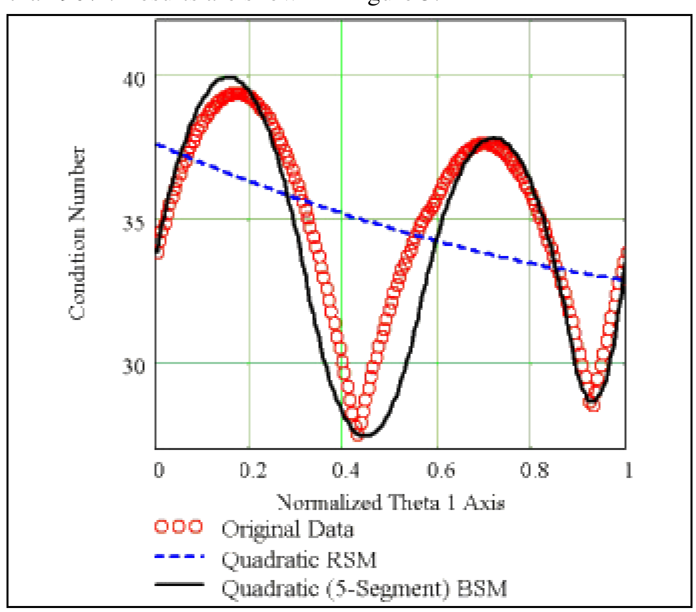

Figure 5. *Case 2: Metamodel Accuracy.*

3.2. 3D RSM VERSUS SPLINE MODELS

The third case is a 3D metamodel, where RWVC is a function of link length 1 and 2. Both the RSM and BSM models fit the resulting data with very little error and more than a 99% correlation. Notably, while the BSM is quadratic along one axis, it is linear along the second axis. The metamodel performance is comparable, but the BSM actually has slightly less computational complexity. Results are shown in Figure 6.

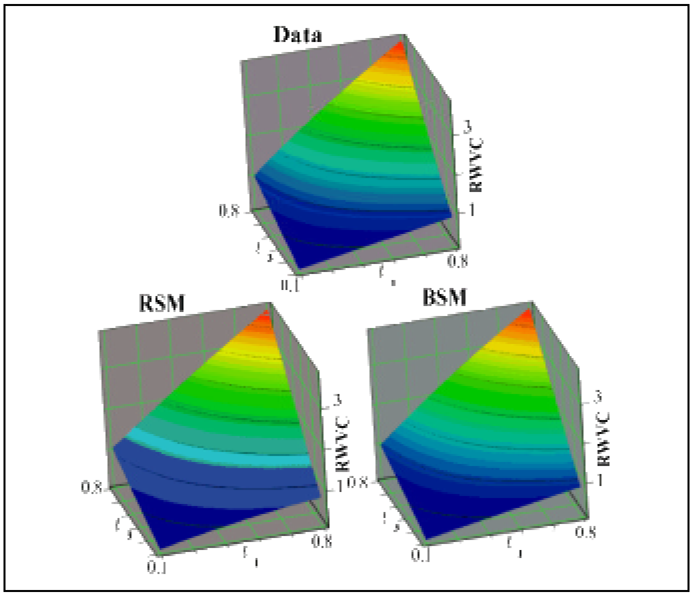

Figure 6. *Case 3: Metamodel Accuracy.*

The fourth case considered was the RWVC under the presence of a geometric boundary. The workspace volume no longer grows with increasing link lengths, but begins to shrink as the robot workspace extends past the boundary. For this application, the RWVC was formulated as a ratio of the robot workspace volume within the geometric boundary with respect to the 2-unit square volume that defines the boundary. The metamodel results are shown in Figure 7.

Without developing supporting software, iterations were limited to the generation of a 25 control point mesh, representing a 3x3 mesh of quadratic patches. At this point, the correlations and maximum errors produced by the two metamodels were equivalent. However, while the narrow "ridgeline" that is apparent in the data is better represented by the BSM, the "3-peak" representation is misleading, and thus the RSM has apparently better visualization properties. Subsequent iterations would probably improve the BSM representation.

The final case considered was for the JCNC, as a function of θ_1 and θ_3, for fixed link lengths of (0.5, 0.4, 0.1) and $\theta_2=30^o$. Like case 2, this is a highly nonlinear function. Figure 8 shows the results.

The BSM used for case 5 was also limited to 25 control point mesh, representing a 3x3 mesh of quadratic patches. Both correlations remain low, 56% for the BSM and 41% for the RSM, and significant errors remain. However, the BSM resembles the data more closely than the RSM. Based on the results from case 2, a mesh of 49 control points could achieve an estimated 90% fit. Additional work is needed to facilitate the calculation of larger control point meshes.

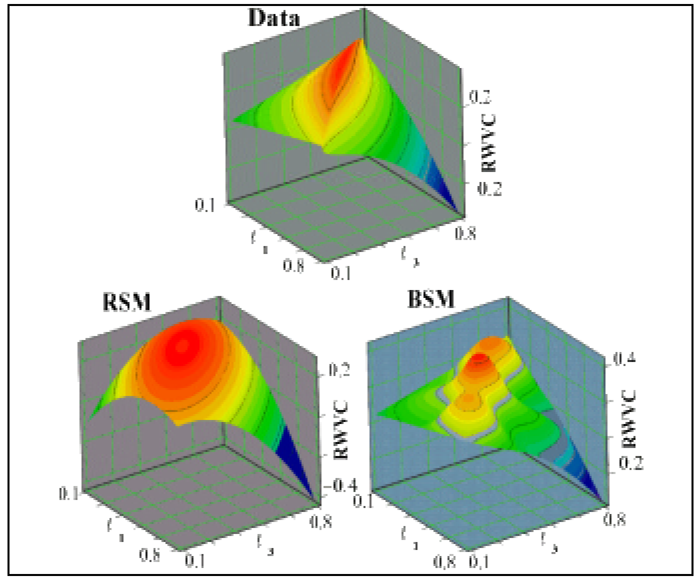

Figure 7. *Case 4: Metamodel Accuracy.*

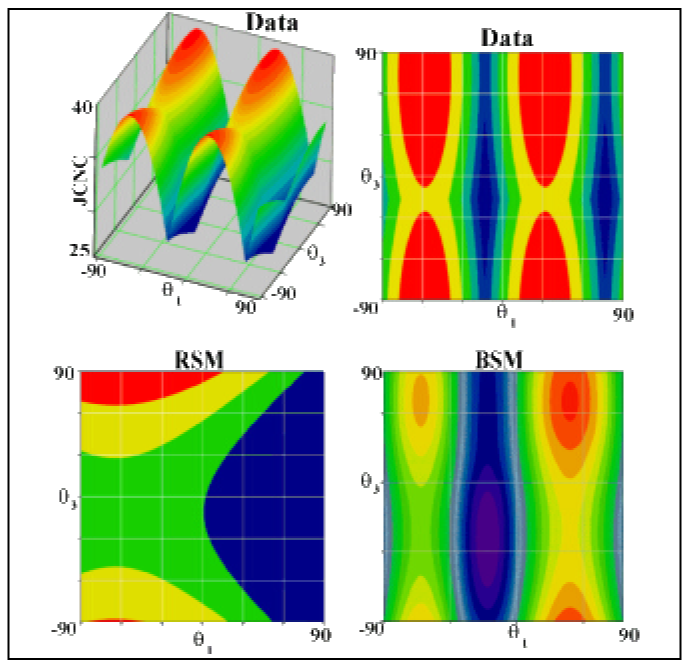

Figure 8. *Case 5: Metamodel accuracy.*

3.3. MODEL PERFORMANCE

Based on the relative performance of the BSM with respect to the RSM in each of the five cases defined in sections 4.1 and 4.2, and the six criteria identified in section 2.5, the Decision Matrix shown in Table 2 was developed. Each trial was rated as successively more difficult and each of the 6 criteria was weighted according to their importance. The BSM was rated in comparison to the RSM baseline.

Table 2. *Decision Matrix comparing RSMs with BSMs for 6 criteria and 5 trial cases.*

Criteria \ Relative Weights \ Demo		Trial #1: 2D Workspace Area		Trial #2: 2D Condition Number		Trial #3: 3D Workspace Area		Trial #4: Bounded 3D Workspace		Trial #5: 3D Condition Number		Summary	
		RSM	BSM	RSM	BSM	RSM	BSM	RSM	BSM	RSM	BSM	RSM	BSM
		0.10	0.10	0.15	0.15	0.20	0.20	0.25	0.25	0.30	0.30	1.00	1.00
Criteria #1: Computational Complexity	0.20	50 / 1.00	50 / 1.00	50 / 1.50	50 / 1.50	50 / 2.00	70 / 2.80	50 / 2.50	50 / 2.50	50 / 3.00	50 / 3.00	10	11
Criteria #2: Model Accuracy	0.30	50 / 1.50	50 / 1.50	50 / 2.25	90 / 4.05	50 / 3.00	50 / 3.00	50 / 3.75	50 / 3.75	50 / 4.50	70 / 6.30	15	19
Criteria #3: Model Insight & Visualization	0.10	50 / 0.50	50 / 0.50	50 / 0.75	90 / 1.35	50 / 1.00	50 / 1.00	50 / 1.25	40 / 1.00	50 / 1.50	70 / 2.10	5	6
Criteria #4:Model Flexibility	0.15	50 / 0.75	80 / 1.20	50 / 1.13	80 / 1.80	50 / 1.50	80 / 2.40	50 / 1.88	80 / 3.00	50 / 2.25	80 / 3.60	8	12
Criteria #5: New Data Incorporation	0.15	50 / 0.75	70 / 1.05	50 / 1.13	70 / 1.58	50 / 1.50	70 / 2.10	50 / 1.88	70 / 2.63	50 / 2.25	70 / 3.15	8	11
Criteria #6: Commercial Availability	0.10	50 / 0.50	0 / 0.00	50 / 0.75	0 / 0.00	50 / 1.00	0 / 0.00	50 / 1.25	0 / 0.00	50 / 1.50	0 / 0.00	5	0
Overall Rating	1.00	5.0	5.3	7.5	10.3	10.0	11.3	12.5	12.9	15.0	18.2	50	58

Because of the complexity of the Decision Matrix, a simplified Pugh chart shown in Table 3 was developed from the Decision Matrix. Five relative performance levels performance were identified, significant improvement (++), improvement (+), equivalent (X), degradation (-), and significant degradation (--), and the BSM was rated using the RSM as a baseline.

Table 3. *Pugh Chart comparing RSMs with BSMs for 6 criteria and 5 trial cases.*

Criteria	Case					Total
	1	2	3	4	5	
1	X	X	+	X	X	+1
2	X	++	X	X	+	+3
3	X	+	X	-	+	+1
4	+	+	+	+	+	+5
5	+	+	+	+	+	+5
6	- -	- -	- -	- -	- -	-10
Total	0	+3	+1	-1	+2	+5

Criterion 1, computational complexity, suggests the techniques are essentially equivalent, based upon Gopi and Manohar's demonstration that a B-spline's computational complexity is determined by its order just as is the case for a polynomial [10]. The case 3 results give a slight edge to BSMs, since their model is equivalent, but is not fully quadratic.

Criterion 2, model accuracy, strongly favors BSMs for case 2, and based on the progress of the mesh in case 5, suggests that some benefit also exists for this trial. This criterion was calculated based on the how well the metamodel fit the data, as does not evaluate the data quality. The accuracy of each trial is shown in Table 4. The development of appropriate software facilitating mesh generation would greatly enhance the ability to examine this concept for problems of realistic complexities.

Criterion 3, model visualization favors BSM in large part because they more accurately represent the data for cases 2 and 5, while favoring the case 4 RSM representation due to the misleading "3-peak" BSM representation. The rating of this criterion is somewhat subjective, but is based on a visual comparison of the actual data to the resulting model in an attempt to penalize a model that may produce an exceptionally accurate fit by inducing more variations than probably actually exist.

Table 4. *Metamodel Accuracy for each trial case. (100% is a perfect fit)*

Model	Case				
	1	2	3	4	5
RSM	99.5%	42.9%	~100%	96.2%	40.9%
BSM	99.5%	95.9%	~100%	96.9%	59.5%

Criterion 4, model flexibility, favors the more flexible geometric BSM representation based on reviews in the literature [10,25]. Criterion 5 favors the local BSM behavior with respect to new data, also based on the literature survey [22]. In this situation, B-Spline basis functions are also more stable than the power basis functions used in RSMs [1]. Criterion 6 strongly favors the present commercial availability of RSM supporting software. In a commercial design setting, this criterion is particularly valid, although in a research setting, it can be argued that it should not be considered.

Overall, BSMs are slightly favored, (more definitively so if criterion 6 is resolved with the development of supporting software), particularly for the nonlinear cases 2 and 5. For the remaining cases, BSMs appear to perform approximately as well as RSMs. This is not surprising since BSMs are a generalization of RSMs. The BSM advantage lies with highly nonlinear applications.

4. CONCLUSIONS AND FUTURE WORK

This comparison does not support a definitive conclusion that BSMs are more suitable than RSMs to solve robot workspace optimization problems. Both techniques are suitable for simple test problems, although BSMs appear to have an edge in representing more complex nonlinearities. One can conclude that BSMs are as suitable as RSMs, and possibly more suitable for nonlinear cases. As this is a preliminary study, these results are sufficiently encouraging to suggest additional work is warranted.

Unfortunately, many algorithms succeed with simple cases only to fail when applied to complex "real" problems [9]. Therefore, additional work examining BSM performance for complex and nonlinear "real" optimization problems is needed. Software tools to support this research also address the primary BSM disadvantage, the current lack of available BSM software. Development of supporting software could verify this apparent performance advantage. Initial results, while promising, require additional research.

This research is a preliminary investigation to determine if a metamodeling method based on spline theory would be competitive with existing approaches such as RSMs. The results are sufficiently promising to justify further research into the potential of BSMs to represent the design space of complex systems. The ultimate goal of this research is the development of a method, based on spline theory that can represent the complex relationships between design variables and performance indices that can be used to enhance the effectiveness of the engineering design process.

ACKNOWLEDGMENTS

This work has been supported by Robert Hollen and Dr. Christopher James of Los Alamos National Laboratory and Dr. Richard Crawford of the University of Texas at Austin. The assistance and encouragement of the rest of the faculty of the Department of Mechanical Engineering at The University of Texas should also be acknowledged. The support of these individuals and organizations are greatly appreciated.

REFERENCES

[1] Barton, R.R., 1994, "Metamodeling: A State-of-the-Art Review," ACM *Proc. of the 1994 Winter Simulation Conference*, Tew, J.D., et al, eds., pp. 237-44.

[2] Barton, R.R., 1998, "Simulation Metamodels," ACM *Proc. of the 1998 Winter Simulation Conference*, Medeiros, D.J., et al, eds., pp. 167-74.

[3] Berner, D.F. & Snyman, J.A., 1999, "The Influence of Joint Angle Constraints on the Optimum Design of a Planar Robot Manipulator Following a Complicated Prescribed Path," *Computers & Mathematics with Applications*, **37**, pp. 111-24.

[4] Crow, E.L., Davis, F.A., & Maxfield, M.W., 1960, *Statistics Manual*, Dover, New York, NY.

[5] Dimas, E., & Briassoulis, D., 1999, *"3D Geometric Modeling Based on NURBS: A Review,"* Advances in Engineering Software, **30**, pp. 741-51.

[6] Doel, K., & Pai, D., 1996, "Performance Measures for Robot Manipulators: A Unified Approach," *The International Journal of Robotics Research*, **15**, pp. 92-111.

[7] Gao, F., & Gruver, W.A., 1997, "Performance Evaluation Criteria for Analysis and Design of Robotic Mechanisms," *Proc. of the 1997 IEEE Int'l Conference on Robotics & Automation*, Albuquerque, NM, pp. 879-84.

[8] Gao, F., Guy, F., & Gruver, W.A., 1997, "Criteria Based Analysis and Design of Three Degree of Freedom Planar Robotic Manipulators," *Proc. of the 1997 IEEE Int'l Conference on Robotics & Automation*, Albuquerque, NM, pp. 468-73.

[9] Gershenfeld, N., 1999, *The Nature of Mathematical Modeling*, Cambridge University Press, Cambridge, UK.

[10] Gopi, M., & Manohar, S., 1997, "A Unified Architecture for the Computation of B-Spline Curves and Surfaces," IEEE *Trans. On Parallel & Distributed Systems*, **8**, pp. 1275-87.

[11] Gosselin, C., & Angeles, J., 1991, "A Global Performance Index for Kinematic Optimization of Robotic Manipulators," ASME *Journal of Mechanical Design*, **113**, pp. 220-6.

[12] Hussain, M.F., Barton, R.R., & Joshi, S.B., 2002, "Metamodeling: Radial Basis Functions, versus Polynomials," *European Journal of Operations Research*, **138**, pp. 142-54.

[13] Kammer, D.C., Alvin, K.A., & Malkus, D.S., 2002, "Combining Metamodels with Rational Function Representations of Discretization Error for Uncertainty Quantification," *Computer Methods in Applied Mechanics & Engineering*, **191**, pp. 1367-79.

[14] Kleijnen, J., & Sargent, R.G., 2000, "A Methodology for Fitting & Validating Metamodels in Simulation," *European J. of Operations Research*, **120**, pp. 14-29.

[15] Klein, C.A., & Blaho, B.E., 1987, "Dexterity Measures for the Design and Control of Kinematically Redundant Manipulators," *The International Journal of Robotics Research*, **6**, pp. 72-83.

[16] Kobbelt, L.P., Botsch, M., Schwanecke, U., & Seidel, H.P., 2001, "Feature Sensitive Feature Extraction from Volume Data," ACM *Proc. SIGGRAPH*, Los Angeles, CA, pp. 57-66.

[17] Legault, J., 2000, "A Complexity Management Framework for Open Architecture Agile Manufacturing Systems," M.S. thesis, The University of Texas at Austin, Austin, TX.

[18] McCarthy, J.M., 2000, "Mechanism Synthesis Theory and the Design of Robots," *Proc. of the 2000 IEEE Int'l Conference on Robotics & Automation*, San Francisco, CA, pp. 55-60.

[19] Nash, S.G., & Sofer, A., 1996, *Linear and Nonlinear Programming*, McGraw Hill, New York, NY.

[20] Park, F.C., & Brockett, R.W., 1994, "Kinematic Dexterity of Robot Mechanisms," *The International Journal of Robotics Research*, **13**, pp. 1-15.

[21] Piegl, L., 1991, "On NURBS: A Survey," IEEE *Computer Graphics & Applications*, pp. 55-71.

[22] Piegl, L., & Tiller, W., 1997, *The NURBS Book, 2nd Ed.*, Springer, Berlin.

[23] Piegel, L., & Tiller, W., 1997, "Symbolic Operators for NURBS," *Journal of Computer-Aided Design*, **29**, pp. 361-8.

[24] Rogers, D.F., & Adams, J.A., 1990, *Mathematical Elements of Computer Graphics, 2nd Ed.*, McGraw Hill, New York, NY.

[25] Shyy, W., Papila, N., Vaidyanathan, R., & Tucker, K., 2001, "Global Design Optimization for Aerodynamics and Rocket Propulsion Components," *Progress in Aerospace Science*, **37**, pp. 59-118.

[26] Simpson, T.W., Peplinski, J.D., Koch, P.N., & Allen, J.K., 1997, "On the Use of Statistics in Design and the Implication for Deterministic Computer Experiments," ASME *1997 Design and Engineering Technical Conferences*, DTM-3881, pp. 1-12.

[27] Simpson, T.W., Peplinski, J.D., Koch, P.N., & Allen, J.K., 2001, "Metamodels for Computer-based Engineering Design:

Survey and Recommendations," *Engineering with Computers*, **17**, pp. 129-50.
[28] Turner, C., 2000, "Criteria Development for Actuator Resource Management," M.S. thesis, The University of Texas at Austin, Austin, TX.
[29] Vijaykumar, R., Waldron, K.J., & Tsai, M.J., 1986, "Geometric of Serial Chain Manipulators for Working Volume and Dexterity," *The International Journal of Robotics Research*, **5**, pp. 91-103.
[30] Wang, N., & Ge, P., 1999, "Study of Metamodeling Techniques and Their Applications in Engineering Design," ASME-MED *Manufacturing Science and Engineering*, **10**, pp. 89-95.
[31] Wang, T., & Wu, K., 1999, "A Parallel Set Design Proceedure for the Simulated Annealing Algorithm Under Computational Time Constraints," *Computers and Operations Research*, **26**, pp. 665-78.
[32] Wu, A., Wu, K.Y., Chen, R.M.M., & Shen, Y., 1998, "Parallel Optimal Statistical Design Method With Response Surface Modeling Using Genetic Algorithms," IEE *Proc. Circuit Device Systems*, **145**, pp. 7-12.
[33] Yoshikawa, T., 1985, "Manipulability of Robotic Mechanisms," *The International Journal of Robotics Research*, **3**, pp. 3-9.