JOURNAL OF PROPULSION AND POWER
Vol. 17, No. 2, March–April 2001

Response Surface and Neural Network Techniques for Rocket Engine Injector Optimization

Wei Shyy*
University of Florida, Gainesville, Florida 32611-6250
P. Kevin Tucker†
NASA Marshall Space Flight Center, Alabama 35812
and
Rajkumar Vaidyanathan‡
University of Florida, Gainesville, Florida 32611-6250

The response surface methodology for rocket engine injector design optimization for which only modest amounts of data may exist is examined. Two main aspects are emphasized: relative performance of quadratic and cubic polynomial response surfaces and enhancement of the fidelity of the response surface via neural networks. A data set of 45 design points from a semi-empirical model for a shear coaxial injector element using gaseous oxygen and gaseous hydrogen propellants is used to formulate response surfaces using quadratic and cubic polynomials. This original data set is also employed to train a two-layered radial basis neural network (RBNN). The trained network is then used to generate additional data to augment the original information available to characterize the design space. Quadratic and cubic polynomials are again used to generate response surfaces for this RBNN-enhanced data set. The response surfaces resulting from both the original and RBNN-enhanced data sets are compared for accuracy. Whereas the cubic fit is superior to the quadratic fit for each data set, the RBNN-enhanced data set is capable of improving the accuracy of the response surface if noticeable errors from polynomial curve fits are encountered. Furthermore, the RBNN-enhanced data set yields more consistent selections of optimal designs between cubic and quadratic polynomials. The techniques developed can be directly applied to injector design and optimization for rocket propulsion.

Nomenclature

A	=	lowest acceptable value of energy release efficiency (ERE)
a	=	radial basis neural network output
B	=	target value of ERE
b	=	bias associated with a neuron in neural networks
C	=	target value of Q
D	=	composite desirability function
d_1	=	desirability function related to ERE
d_2	=	desirability function related to Q
E	=	highest acceptable value of Q
e_i	=	error at the ith design point
L_{comb}	=	combustor length (length from injector to throat)
n	=	number of data points
n_p	=	number of coefficients in the response surface
O/F	=	propellant mixture ratio
$\boldsymbol{p}$	=	input vector of the neural network
Q	=	actual chamber wall heat flux
Q_{nom}	=	nominal chamber wall heat flux
radbas	=	transfer function of radial basis neural network
s	=	weighting factor for d_1
t	=	weighting factor for d_2
σ	=	root mean square error
σ_a	=	adjusted root mean square error

Received 15 June 1999; revision received 10 August 2000; accepted for publication 13 August 2000.

*Professor, Department Chair, Department of Aerospace Engineering, Mechanics, and Engineering Science.
†Aerospace Technologist, Space Transportation Directorate, NASA Marshall Space Flight Center.
‡Graduate Student, Department of Aerospace Engineering, Mechanics, and Engineering Science.

I. Introduction

THE injector design methodologies used successfully in previous rocket propulsion system development programs were typically based on large subscale databases and the empirical design tools derived from them.[1–5] Extensive sub- and full-scale hot-fire test programs often guided these methodologies. Current and planned launch vehicle programs have tight budgets and aggressive schedules, neither of which is conductive to the large test programs of the past. Also, new requirements for operability and maintainability require that injector design be robust. Hence, variables not previously included in the injector design now merit consideration for inclusion in the design process. Also, the effect of the injector design on variables, peripheral to, but influenced by the injector, may need to be included in the injector design process. These new programs with compressed schedules, lower budgets, and more stringent requirements make development of broader and more efficient injector design methodologies an attractive goal.

Historically, injectors have been designed, fabricated, and tested based on experience and intuition. As hardware was tested, designers proposed modifications aimed at obtaining an improved design. Despite their experience and skill, these efforts were unlikely to produce the optimal design in a short time frame. Also, as more design variables are considered, the design process becomes increasingly complex, and it is more difficult to foresee the effect of the modification of one variable on other variables. Use of an optimization approach to guide the design addresses both of these issues. The optimization scheme allows complex, interrelated information to be managed in such a way that the extent to which variables influence each other can be objectively evaluated and optimal design points can be identified with confidence.

Development of an optimization scheme for injector design called methodology for optimizing the design of injectors (*method I*) has been reported by Tucker et al.[6] *Method I* is used to generate appropriate injector design data and then guide the designer toward an optimum design subject to the specified constraints. As reported,

method I uses the response surface methodology (RSM) to facilitate the optimization. The RSM approach is to conduct a series of well-chosen experiments (empirical, numerical, physical, or some combination of the three) and use the resulting information to construct a global approximation (response surface) of the measured quantity (response) over the design space. A standard constrained optimization algorithm is then used to interrogate the response surface for an optimum design.

The initial demonstration of *method I* by Tucker et al.[6] focused on a simple optimization of a shear coaxial injector element with gaseous oxygen and gaseous hydrogen propellants. The design data were generated using an empirical design methodology developed by Calhoon et al.[7] These researchers conducted a large number of cold-flow and hot-fire tests over a range of propellant mixture ratios, propellant velocity ratios, and chamber pressure for shear coaxial, swirl coaxial, impinging, and premixed elements. The data were correlated directly with injector/chamber design parameters, which are recognized from both theoretical and empirical standpoints as the controlling variables. For the shear coaxial element, performance, as measured by energy release efficiency (*ERE*), is obtained using correlations taking into account combustor length L_{comb} (length from injector to throat) and the propellant velocity ratio V_f/V_0. The nominal chamber wall heat flux at a point just downstream of the injector, Q_{nom}, is calculated using a modified Bartz equation and is correlated with propellant mixture ratio O/F and propellant velocity ratio V_f/V_0 to yield the actual chamber wall heat flux Q. The objective in the initial demonstration of *method I* was to maximize injector performance while minimizing chamber wall heat flux (lower heat fluxes reduce cooling requirements and increase chamber life) and chamber length (shorter chambers lower engine weight).

The initial demonstration of *method I* used quadratic polynomials to generate the response surfaces. The surfaces for *ERE* and Q were joined by use of a desirability function, and optimum design points were sought as the independent variables (O/F, V_f/V_0, and L_{comb}) were constrained over different ranges. The initial demonstration reported by Tucker et al.[6] is viewed as a proof of concept for *method I*.

II. Scope of Current Research

Empirical design methodologies, such as that by Calhoon et al.,[7] may allow the designer to generate large quantities of data within a design space. However, due to their empiricism, these methodologies are often sufficiently accurate only over the range of variables for which test data were taken to develop the methodology. For some injector types, propellant combinations, or design conditions, this limitation may require that additional data be generated to ensure confidence in the design. Historically, these data have been generated in sub- and full-scale test programs. More recently computational fluid dynamics (CFD) analysis from validated models has been used to augment the test data. Data from test programs and CFD analysis are expensive and time consuming to obtain. Recognition of this has direct implications for the usefulness of optimization techniques in injector design methodologies. Although the optimization scheme must be capable of efficiently organizing large amounts of design information generated from empirical design methodologies, it must also be able to make effective use of the relatively small amounts of data available in some cases. An optimization scheme that requires large amounts of data to generate meaningful results will be marginally useful, if at all, when only small amounts of data are available for use.

The present effort seeks to investigate approaches that would make RSM robust and reliable for injector design optimization, especially when only limited amounts of design data exist. We first investigate the relative performance of a quadratic and a cubic polynomial for constructing response surfaces. The original data set from Tucker et al.[6] (with 45 design points) is used to generate the response surfaces for *ERE* and Q. The quality of each fit on the original data set is evaluated. Then, an approach to train a radial basis neural network (RBNN) to enhance the information available to construct the response surface is presented. Specific issues relative to the network training are evaluated and discussed. This trained RBNN is then used to generate additional design data. The data generated from the network are combined with the original data from the Calhoon et al.[7] model to form an enhanced data set, which is then refit with quadratic and cubic polynomial surfaces. The quality of the fit of the resulting surfaces is compared. Also, each surface is used to conduct design optimization over the same range of independent variables. The optimal design points are compared with exact points calculated from the empirical model.

III. Approaches

A. General

The range of propellant mixture ratios O/F propellant velocity ratios V_f/V_0, and chamber lengths L_{comb} considered in this study are shown in Table 1. Tables 2–4 shows the empirically derived performance and heat flux for the 45 combinations of O/F, V_f/V_0, and L_{comb} considered. Hereafter, these 45 design points are referred to as the original data set. As noted earlier, this original data set is augmented with additional design points generated using a trained RBNN. This new and larger data set is referred to as the enhanced data set. The RSM, using both quadratic and cubic polynomials, is used to fit both surfaces. The following two sections give pertinent details on the RSM and neural networks (NN).

Table 1 Range of design variables considered

O/F	V_f/V_0	L_{comb}, in.
4, 6, 8	4	4–8
4, 6, 8	6	4–8
4, 6, 8	8	4–8

Table 2 Performance and heat flux responses for *O/F* = 4 elements

O/F	V_f/V_0	L_{comb}, in.	*ERE*, %	Q, Btu/in.^{2}-s
4.0	4.0	4.0	92.9	0.753
4.0	4.0	5.0	96.0	0.753
4.0	4.0	6.0	97.6	0.753
4.0	4.0	7.0	98.6	0.753
4.0	4.0	8.0	99.0	0.753
4.0	6.0	4.0	95.0	0.928
4.0	6.0	5.0	97.1	0.928
4.0	6.0	6.0	98.5	0.928
4.0	6.0	7.0	99.2	0.928
4.0	6.0	8.0	99.4	0.928
4.0	8.0	4.0	96.6	1.10
4.0	8.0	5.0	98.2	1.10
4.0	8.0	6.0	99.1	1.10
4.0	8.0	7.0	99.4	1.10
4.0	8.0	8.0	99.6	1.10

Table 3 Performance and heat flux responses for *O/F* = 6 elements

O/F	V_f/V_0	L_{comb}, in.	*ERE*, %	Q, Btu/in.^{2}-s
6.0	4.0	4.0	92.9	0.691
6.0	4.0	5.0	96.0	0.691
6.0	4.0	6.0	97.6	0.691
6.0	4.0	7.0	98.6	0.691
6.0	4.0	8.0	99.0	0.691
6.0	6.0	4.0	95.0	0.642
6.0	6.0	5.0	97.1	0.642
6.0	6.0	6.0	98.5	0.642
6.0	6.0	7.0	99.2	0.642
6.0	6.0	8.0	99.4	0.642
6.0	8.0	4.0	96.6	0.741
6.0	8.0	5.0	98.2	0.741
6.0	8.0	6.0	99.1	0.741
6.0	8.0	7.0	99.4	0.741
6.0	8.0	8.0	99.6	0.741

Table 4 Performance and heat flux responses for *O/F* = 8 elements

O/F	V_f/V_0	L_{comb}, in.	*ERE*, %	Q, Btu/in.^{2}-s
8.0	4.0	4.0	92.9	0.588
8.0	4.0	5.0	96.0	0.588
8.0	4.0	6.0	97.6	0.588
8.0	4.0	7.0	98.6	0.588
8.0	4.0	8.0	99.0	0.588
8.0	6.0	4.0	95.0	0.512
8.0	6.0	5.0	97.1	0.512
8.0	6.0	6.0	98.5	0.512
8.0	6.0	7.0	99.2	0.512
8.0	6.0	8.0	99.4	0.512
8.0	8.0	4.0	96.6	0.493
8.0	8.0	5.0	98.2	0.493
8.0	8.0	6.0	99.1	0.493
8.0	8.0	7.0	99.4	0.493
8.0	8.0	8.0	99.6	0.493

B. RSM

The approach of RSM[8] is to perform a series of experiments, or numerical analyses, for a prescribed set of design points, and to construct a response surface of the measured quantity over the design space. In the present context, the two responses of interest are a measure of combustor performance *ERE* and the injector wall heat flux Q. The design space consists of the set of relevant variables O/F, V_f/V_0, and L_{comb} considered over the ranges shown in Table 1. The response surfaces are fit by standard least-squares regression with a quadratic polynomial using JMP[9] statistical analysis software. JMP is an interactive, spreadsheet-based program that provides a variety of statistical analysis functions. A backward elimination procedure based on t-statistics is used to discard terms and improve the prediction accuracy. The t-statistic, or t-ratio, of a particular coefficient is given by the value of the coefficient divided by the standard error of the coefficient. The quality of fit between different surfaces can be evaluated by comparing the adjusted rms error defined as

$$\sigma_a = \sqrt{\sum e_i^2 / (n - n_p)} \tag{1}$$

where σ_a is the adjusted rms error incurred while mapping the surface over the data set. The measure of error given by σ_a is normalized to account for the degrees of freedom in the model. This rms error, thus, accounts for the nominal effect of higher-order terms providing a better overall comparison among the different surface fits.

In the current study, it is desirable to simultaneously maximize *ERE* and minimize Q. One method of optimizing multiple responses simultaneously is to build, from the individual responses, a composite response known as the desirability function. The method allows for a designer's own priorities on the response values to be built into the optimization procedure. The first step in the method is to develop desirability function d for each response. In the case where a response should be maximized, such as *ERE*, the desirability takes the form

$$d_1 = [(ERE - A)/(B - A)]^s \tag{2}$$

where A is the lowest acceptable value such that $d = 1$ for any $ERE > B$ and $d = 0$ for $ERE < A$. The power values s is a weighting factor, which is set according to one's subjective impression about the role of the response in the total desirability of the product. In the case where a response is to be minimized, such as Q, the desirability takes on the form

$$d_2 = [(Q - E)/(C - E)]^t \tag{3}$$

where E is the highest acceptable value such that $d = 1$ for any $Q < C$ and $d = 0$ for $Q > E$. Choices for *A, B, C,* and E are made according to the designer's priorities or, as in the present study, simply as the boundary values of the domain of *ERE* and Q spanned by the points in Tables 2–4. Values of s and t are set based on which response has higher priority. A single composite response is developed that is the geometric mean of the desirabilities of the individual responses. The composite response defined as

$$D = (d_1 \cdot d_2 \cdot d_3 \cdots d_m)^{1/m} \tag{4}$$

which for the present case is

$$D = (d_1 \cdot d_2)^{\frac{1}{2}} \tag{5}$$

This is then submitted to an optimization toolbox to be maximized. Solver, an optimization tool available as part of Microsoft Excel[10] package, is used in this effort. This tool uses the GRG2 nonlinear optimization code developed by Lasdon et al.[11]

C. NN

NN[12,13] have received much attention in engineering applications in the last decade because they are hihgly flexible and have the ability to be trained, using user-supplied data, to map complex surfaces. The NN can be trained with data from any source: empirical, experimental, or analytical. Training is accomplished by adjusting weights on the internal connections of the network through defined training algorithms. The training is a cyclic process in which the weights and biases are repeatedly adjusted until an accurate mapping of the response data is obtained. Once trained, the NN is then able to predict the responses for other points in the design space. The NN toolbox available in MATLAB®[14] is used in the current work. A two-layered radial basis network is used to provide the mapping between the input parameters (independent variables) and the output parameters (dependent variables). The network in this effort is designed with the function Solverbe and simulated with Simurb, both of which are contained in the MATLAB NN toolbox. Solverbe designs the network with zero error on the training vectors. It uses the known set of inputs and target vectors along with a quantity called the spread constant to generate the weights and biases for the exact mapping of the network. The designed network has two layers: an initial radial basis layer and a final linear layer. In the initial layer, Solverbe creates as many radial basis neurons as there are input vectors. Each neuron is assigned a weight that is set to the transpose of a given input vector. By design, each neuron detects and responds to a different input vector. Hence, there are as many neurons as input vectors. The radial basis function (*radbas*) has a maximum output of 1 when the input is 0. The radial basis output a is given by

$$a = radbas\{dist(\boldsymbol{w}, \boldsymbol{p}) \times b\} \tag{6}$$

where *radbas* is the transfer function, *dist* is the vector distance between its weight vector $\boldsymbol{w}$ and the input vector $\boldsymbol{p}$, and b is the bias. The bias controls the sensitivity of the neuron. The output has an inverse relationship with the distance between the vectors $\boldsymbol{p}$ and $\boldsymbol{w}$. Any neuron in the network with input identical to its weight vector has an output value of 1. For an input of 0.8326, radial basis produces an output of 0.5. To obtain an output of 0.5 or more, the vector distance between an input vector and its weight vector must be $0.8326/b$ or less. Each bias is set to a value of $0.8326/sc$, where sc is the spread constant. The sc, therefore, defines the range within which the input vector has to lie relative to the weight vector to produce an output of 0.5 or more from the radbas. For large values of sc, neurons should respond strongly to the overlapping regions of the design space. Note that caution must be used regarding selection of values for sc. If the sc value is too large, all neurons may respond to a given input. This creates an erroneous signal, which may adversely affect the network's ability to predict new design points. As discussed in a later section, a study has been conducted to estimate the best value of sc for the present work on injector optimization. Based on the output from the radbas, the second linear layer of the network attempts to map it to the output while minimizing the sum-squared

error. Weights and biases are assigned to each neuron based on its output from the radial basis layer such that the network yields a value sufficiently close to the target vector. After the weights and biases have been generated, the MATLAB function Simurb is employed. This function uses the weights and biases generated by Solverbe during the training period to predict the output for new sets of inputs.

The values of independent and dependent variables in Tables 2–4, which constitute the original data set, are used as the inputs and outputs to train the NN. This trained network is then used to generate other design points as required for the enhanced data set.

The design function and design parameters have been found to play an important role in the design of an efficient network. An attempt was made to study the effect of design function. In addition to Solverbe, another MATLAB design function, Solverb, was used to repeat the training/prediction procedure described. A significant difference was noticed in the prediction capabilities of the two functions. As compared to Solverbe, Solverb adds one neuron at a time instead of adding as many neurons as the number of inputs. It is an iterative procedure, and neurons are added until the error during training is less than the user-defined error goal. This iterative procedure of adding neurons one at a time may result in a smaller network, but it takes a longer time to train than Solverbe. It also requires more design parameters than Solverbe.

More detailed discussion of the basic concepts and practical implementation of both RSM and NN, directly relevant to the present effort, can be found in Ref. 13.

IV. Results and Discussion

A. Polynomial Fits on the Original Data Set

According to the injector model developed by Calhoon et al.,[7] injector performance, as measured by *ERE* depends only on the velocity ratio V_f/V_0 and combustion chamber length L_{comb}. Examination of the original data set in Tables 2–4 indicates 15 distinct design points for *ERE*. Because chamber wall heat flux is dependent only on velocity ratio V_f/V_0 and oxidizer to fuel ratio O/F, there are nine distinct design points for Q. The design space for this effort is shown in Fig. 1. For *ERE*, the five distinct chamber lengths offer the potential for a fourth-order polynomial fit in L_{comb}, whereas the three different velocity ratios limit the fit in V_f/V_0 to second order. Quadratic and cubic response surfaces for both *ERE* and Q has been generated for evaluation. These noted limitations on the data cause the cubic surfaces to be third order in L_{comb} only.

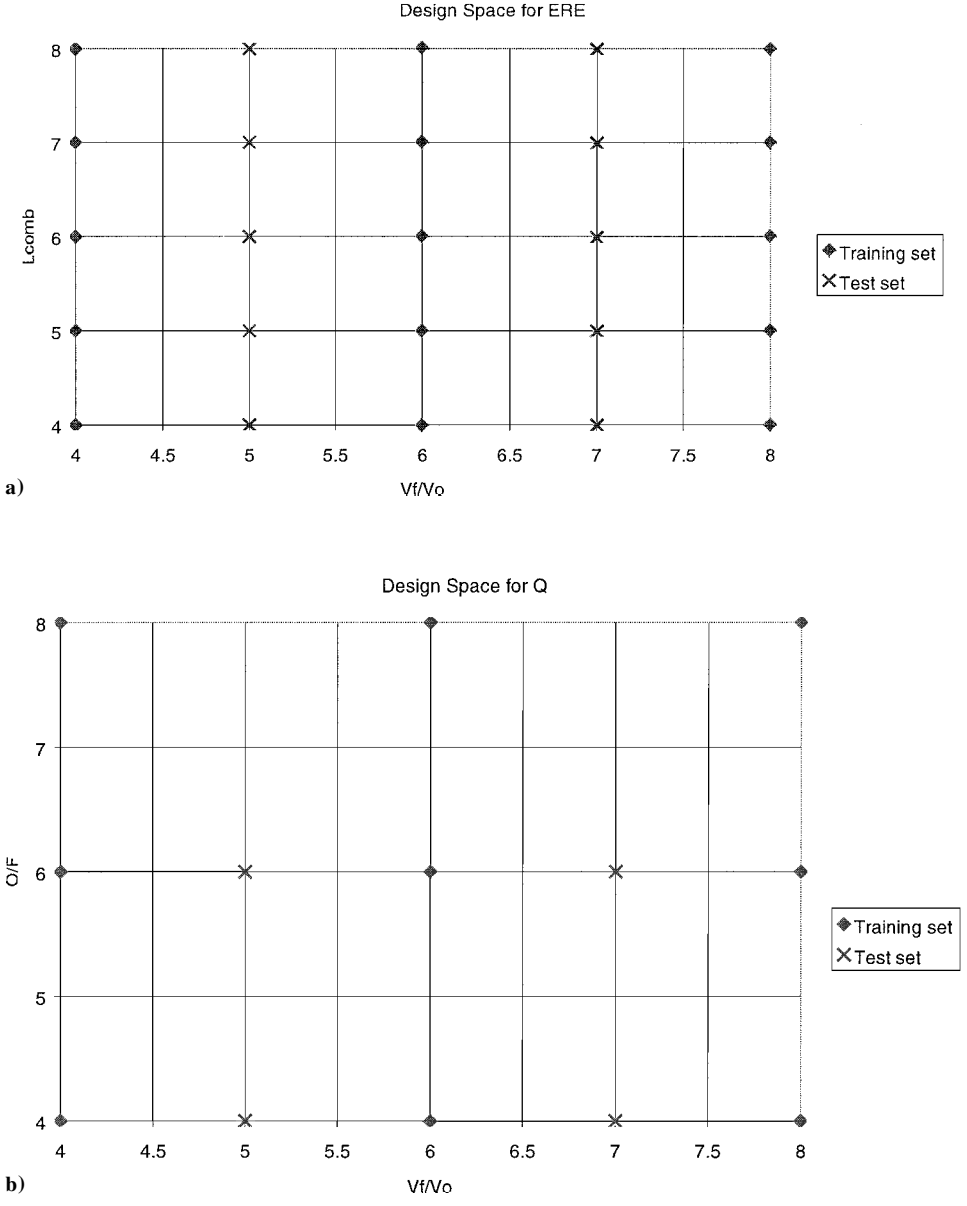

Fig. 1 Design space for a) *ERE*, 15 training/mapping points, and 10 test points and b) Q, 9 training/mapping points and 4 test points.

1. Quadratic Response Surface

The quadratic response surfaces on the original data set constructed by JMP are given by

$$ERE = 70.43 + 1.580V_f/V_0 + 6.208L_{\text{comb}} - 0.190(V_f/V_0)L_{\text{comb}} - 0.331(L_{\text{comb}})^2 \quad (7)$$

and

$$Q = 0.479 - 0.046O/F + 0.191V_f/V_0 + 0.0094(O/F)^2 - 0.028(O/F)V_f/V_0 \quad (8)$$

These response surfaces represent reduced models accomplished by term elimination from the full surface using t statistics as described earlier. These are identical to reduced surfaces generated previously by Tucker et al.[6]

2. Cubic Response Surface

Cubic models for *ERE* and Q have also been generated for the analysis. The full *ERE* and Q response surfaces are

$$ERE = 50.060 + 3.759V_f/V_0 + 14.574L_{\text{comb}} - 0.05(V_f/V_0)^2 - 0.777(V_f/V_0)L_{\text{comb}} - 1.459(L_{\text{comb}})^2 + 0.0025(V_f/V_0)^2L_{\text{comb}} + 0.0464V_f/V_0(L_{\text{comb}})^2 + 0.0472(L_{\text{comb}})^3 \quad (9)$$

and

$$Q = -0.566 - 0.358O/F + 0.383V_f/V_0 - 0.0191(O/F)^2 - 0.107(O/F)V_f/V_0 - 0.0028(V_f/V_0)^2 + 0.0048(O/F)^2V_f/V_0 + 0.0019(O/F)(V_f/V_0)^2 \quad (10)$$

Reduced cubic surfaces for both *ERE* and Q were also obtained. As discussed later, shortcomings were encountered with these surfaces.

3. Comparison Between Cubic and Quadratic Response Surfaces

The quadratic and cubic fits for both surfaces are plotted along with the actual data from the injector model in Fig. 2. NN and injector model data are the same points in the graph. Quadratic and cubic are predicted by RSM. Based on the adjusted rms error, the cubic fit is more accurate than the quadratic fit for *ERE*. The adjusted rms error for the quadratic and cubic response surfaces of *ERE* are 0.211 and 0.083, respectively. The cubic fit, by this measure, is superior for *ERE*. However, the error is almost identical in the case of Q for both the quadratic (0.039) and cubic (0.040) surfaces. This maybe due to the very small number of points available for the curve fit. The additional terms in the cubic fit relative to the quadratic fit do not improve the mapping of the response surface for Q.

In the report by Tucker et al.,[6] an optimization was done for three different ranges of the independent variables using the quadratic fit shown in Eqs. (7) and (8). The three cases analyzed differ only in the constraints implemented on the design parameters. The

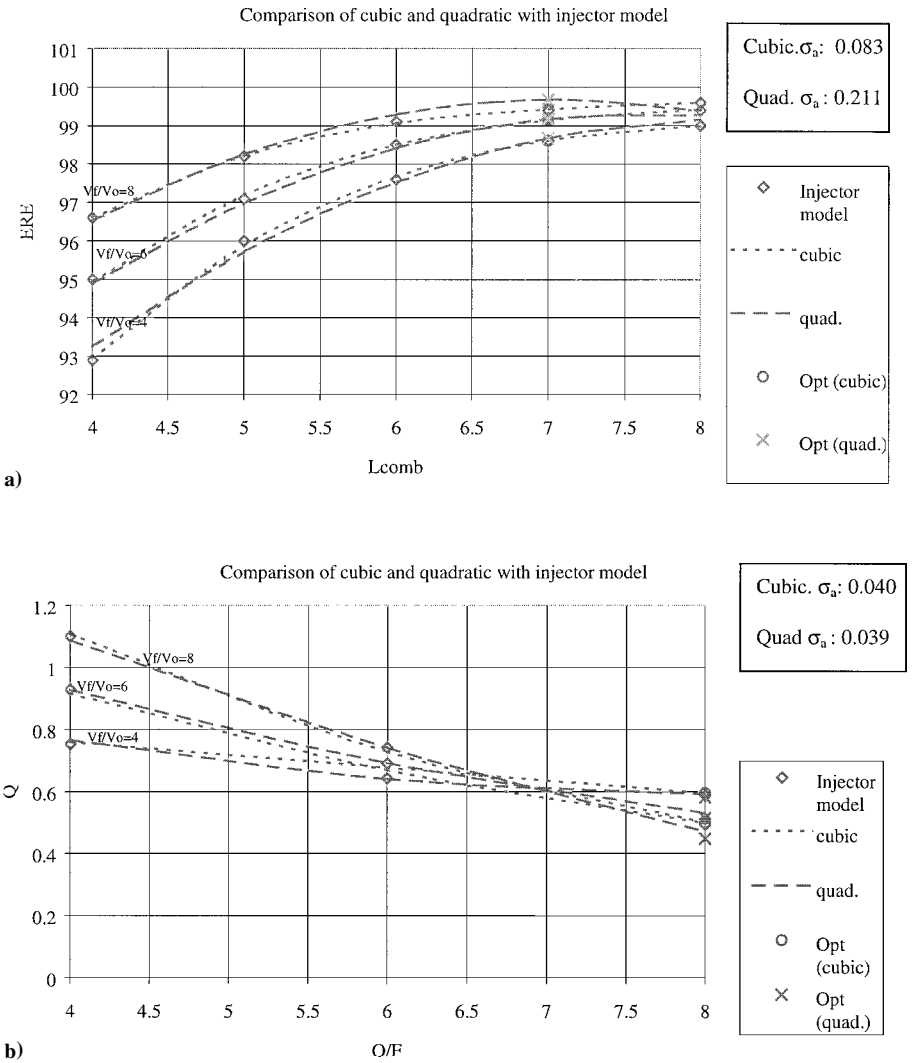

Fig. 2 Assessment of performance of cubic and quadratic response surfaces of a) *ERE*, 15 training/mapping points and b) Q, 9 training/mapping points.

Table 5 Optimum values obtained with cubic and quadratic for case 1[a]

		Cubic					Quadratic				
W_{ERE}, (s)	W_Q, (t)	O/F	V_f/V_0	L_{comb}, in.	*ERE*, %	Q, Btu/in.^{2}-s	O/F	V_f/V_0	L_{comb}, in.	*ERE*, %	Q, Btu/in.^{2}-s
1	10	6.0	**5.41**	7.0	99.02 (99.00)[b]	0.664 (0.654)	6.0	**6.00**	7.0	99.17 (99.20)	0.669 (0.642)
1	1	6.0	6.00	7.0	99.15	0.669	6.0	6.00	7.0	99.17	0.669
10	1	6.0	6.00	7.0	99.15	0.669	6.0	6.00	7.0	99.17	0.669

[a]Constraints: $4 \le O/F \le 6$, $4 \le V_f/V_0 \le 6$, and L_{comb} (in.) ≤ 7. [b](Exact response of the injector model.)

Table 6 Optimum values obtained with cubic and quadratic for case 2[a]

		Cubic					Quadratic				
W_{ERE}, (s)	W_Q, (t)	O/F	V_f/V_0	L_{comb}, in.	*ERE*, %	Q, Btu/in.^{2}-s	O/F	V_f/V_0	L_{comb}, in.	*ERE*, %	Q, Btu/in.^{2}-s
1	10	6.0	**5.41**	7.0	99.02 (99.00)[b]	0.664 (0.654)	6.0	**6.52**	7.0	99.31 (99.10)	0.684 (0.716)
1	1	6.0	**6.34**	7.0	99.21 (99.20)	0.674 (0.691)	6.0	**7.00**	7.0	99.42 (99.30)	0.702 (0.728)
10	1	6.0	7.00	7.0	99.32	0.690	6.0	7.00	7.0	99.42	0.702

[a]Constraints: $4 \le O/F \le 6$, $5 \le V_f/V_0 \le 7$, and L_{comb} (in.) ≤ 7. [b](Exact response of the injector model.)

Table 7 Optimum values obtained with cubic and quadratic for case 3[a]

		Cubic					Quadratic				
W_{ERE}, (s)	W_Q, (t)	O/F	V_f/V_0	L_{comb}, in.	*ERE*, %	Q, Btu/in.^{2}-s	O/F	V_f/V_0	L_{comb}, in.	*ERE*, %	Q, Btu/in.^{2}-s
1	10	6.0	**6.00**	7.0	99.15	0.669	6.0	**6.52**	7.0	99.31	0.684
1	1	6.0	**6.34**	7.0	99.21	0.674	6.0	**8.00**	7.0	99.67	0.753
10	1	6.0	8.00	7.0	99.42	0.728	6.0	8.00	7.0	99.67	0.753

[a]Constraints: $4 \le O/F \le 6$, $6 \le V_f/V_0 \le 8$, and L_{comb} (in.) ≤ 7.

constraints are as follows: case 1, $4 \le O/F \le 6$, $4 \le V_f/V_0 \le 6$, and $L_{comb} \le 7$; case 2, $4 \le O/F \le 6$, $5 \le V_f/V_0 \le 7$, and $L_{comb} \le 7$; and case 3, $4 \le O/F \le 6$, $6 \le V_f/V_0 \le 8$, and $L_{comb} \le 7$.

In the current effort, the optimization is repeated using the cubic fits in Eqs. (9) and (10). The combinations of weights for ERE (s) and Q (t) used are (1,10), (1,1), and (10,1) for each of the three cases. The optimum has been evaluated and tabulated for each case for each of the three weightings. Tables 5–7 show the results for the 18 resulting optimization exercises. Recall, that in this effort, injector element optimization means maximizing the performance while minimizing heat flux and chamber length. The optimum value for V_f/V_0 obtained on the cubic response surface is quite different than that found on the quadratic surface for some cases (these particular cases are noted in bold face in Tables 5–7). Also, for selected cases where there are discrepancies between the quadratic and cubic results, the exact values from the injector model have been included in parentheses in Tables 5–7 for comparison. In these cases, the cubic fit more closely matches the exact data than does the quadratic fit. Sample results for *ERE* plotted in Fig. 3a clearly show the data are better fit by the cubic surface for the case shown. Figure 3b shows that the response surface predicted by cubic fit for Q has a noticeable dip that is completely missed by the quadratic fit. This discrepancy results in the optimum for the cubic fit being considerably lower than that for the quadratic surface. The prediction from cubic fit agrees well with the exact data, which also has a dip for this specific case. NN and injector model data are the same points in Fig. 3. Quadratic and cubic are predicted by RSM.

The injector model was also used to produce additional design points to assess the capability of the different response surfaces to match the exact data. In Figs. 4a and 5a, the actual data obtained from the injector model for all of the design points are shown. The cubic and quadratic response surfaces obtained based on the original data is also shown. The rms error σ for the test data is given by

$$\sigma = \sqrt{e_i^2/n} \tag{11}$$

In this equation, n is the number of test points. The rms error for predicting the new *ERE* data is 0.270 and 0.205 for the quadratic and cubic surfaces, respectively. For Q, it is 0.025 and 0.016 for the quadratic and cubic surfaces, respectively. Again, the performance of the cubic surface is superior to that of the quadratic surface.

A reduced cubic model was also obtained, but the difference in the adjusted rms error was small as compared to the full model. It was found to be 0.077 for *ERE* and 0.042 for Q as compared to the value of 0.083 for *ERE* and 0.040 for Q for the full model. Despite being comparable to the full model from the standpoint of the adjusted rms error, it was found that the rms error in predicting the new data was significantly degraded for Q. The rms error in predicting the new data was 0.203 for *ERE* and 0.038 for Q for the reduced models as compared to 0.205 for *ERE* and 0.016 for Q for the full cubic model. Accordingly, the full model was preferred to the reduced one.

B. RBNN

RBNN are trained by both Solverbe and Solverb for each injector design response, *ERE* and Q, using the original data set of 45 design points. Solverbe trained the network for *ERE* with an error to the order of 10^{-13}. The network trained by Solverbe for Q has an error on the order of 10^{-16}. Both networks represent the respective design spaces essentially exactly. Solverb, with an error goal of 0.001, trained networks for both responses to represent the original data set adequately. Because the size of the data set considered for training the network is fairly modest, the number of neurons generated by Solverbe is also small. Solverb would have been suited better for a larger data set, where a reduction in the number of neurons might have appreciably reduced the computation time. The networks trained using Solverbe have been used for this study. The ability of the RBNN to fit the design data and generate additional data is discussed in the following sections.

1. *Comparison Between Solverbe and Solverb*

Because Solverbe trains with the same number of neurons (45 in this case) as data points, as seen earlier, it fits the training data set

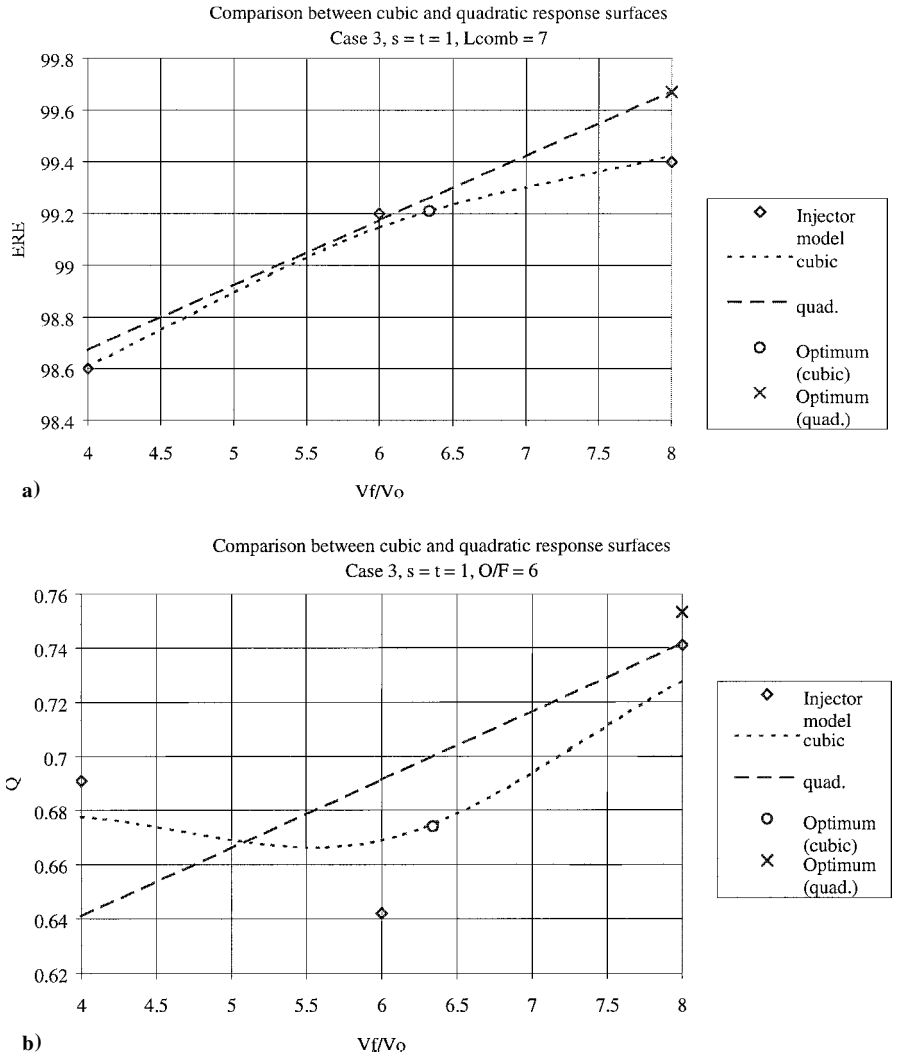

Fig. 3 Comparison between cubic and quadratic response surface for case 3 of a) *ERE* and b) *Q*.

with negligible error. However, it can also create erratic behavior because it makes no attempt to filter noise generated by excess neurons in the network. Solverb, on the other hand, tends to reduce the potential for noise by controlling the number of neurons in the network. Table 8 shows that in the present study, for the spread constant value of 1.00, Solverb performs slightly better than Solverbe based on the nominal error measure. However, when judged by the level of errors associated, both RBNNs are satisfactory from a practical standpoint. As expected, Solverb uses fewer neurons than Solverbe, in this case three less. Note that, as investigated in detail by Papila et al.,[15] the relative performance between Solverb and Solverbe is case dependent.

Table 8 RMS error in the prediction of ERE and *Q*. Different values of SC

	Solverbe		Solverb[a]		
sc	rms error (*ERE*, %)	rms error (*Q*, Btu/in.^{2}-s)	rms error (*ERE*,%)	rms error (*Q*, Btu/m.^{2}-s)	No. of neurons
0.50	1.493	0.179	1.733	0.287	44
0.75	0.745	0.135	0.675	0.135	44
1.00	**0.152**	**0.022**	**0.153**	**0.017**	**42**
1.05	**0.190**	**0.011**	**0.128**	**0.012**	**44**
1.25	0.316	0.010	0.267	0.022	44
1.50	0.336	0.022	0.309	0.030	44
1.75	0.369	0.022	0.310	0.021	44
2.00	0.308	0.016	0.296	0.019	41
2.25	0.279	0.020	1.846	0.045	43
2.50	0.325	0.017	0.744	0.025	43

[a]Error goal used for Solverb is 0.001.

2. *Comparison of RBNN Predictions with RSM*

Figures 4b and 5b show that the RBNN trained by Solverbe is able to generate more accurately additional design data than either quadratic or cubic polynomial (shown for comparison in Figs. 4a and 5a). In Fig. 4a, the *ERE* surface trained with the original data set is shown. The 10 extra design points calculated with the injector model for V_f/V_0 of 5.00 and 7.00 are shown. The ability of the RBNN to generate accurately new design data can be seen by comparing the fit for *ERE* in Fig. 4b to that for the polynomials in Fig. 4a. RBNN trains the network with more flexibility and learns the data trend, whereas polynomials provide only an approximate fit on the given data. Regarding the rms error σ, for *ERE*, it is 0.152 for RBNN predictions as compared to the values of 0.270 and 0.205 for quadratic and cubic surfaces, respectively. The four extra design points generated for *Q*, also at V_f/V_0 of 5.00 and 7.00, are shown compared to the polynomial surface in Fig. 4b and compared to the RBNN surface in Fig. 5b. The rms error in the case of *Q* is 0.022 for RBNN as compared to 0.025 and 0.016 for quadratic and cubic surfaces, respectively. Here the performance of the RBNN is better than the quadratic but slightly poorer than the cubic fit. Examination of Table 8 indicates it may be possible that using Solverb with a spread constant of 1.05 could further reduce the rms for *Q*. However, the errors for *Q* are low enough that further reduction may not be practical.

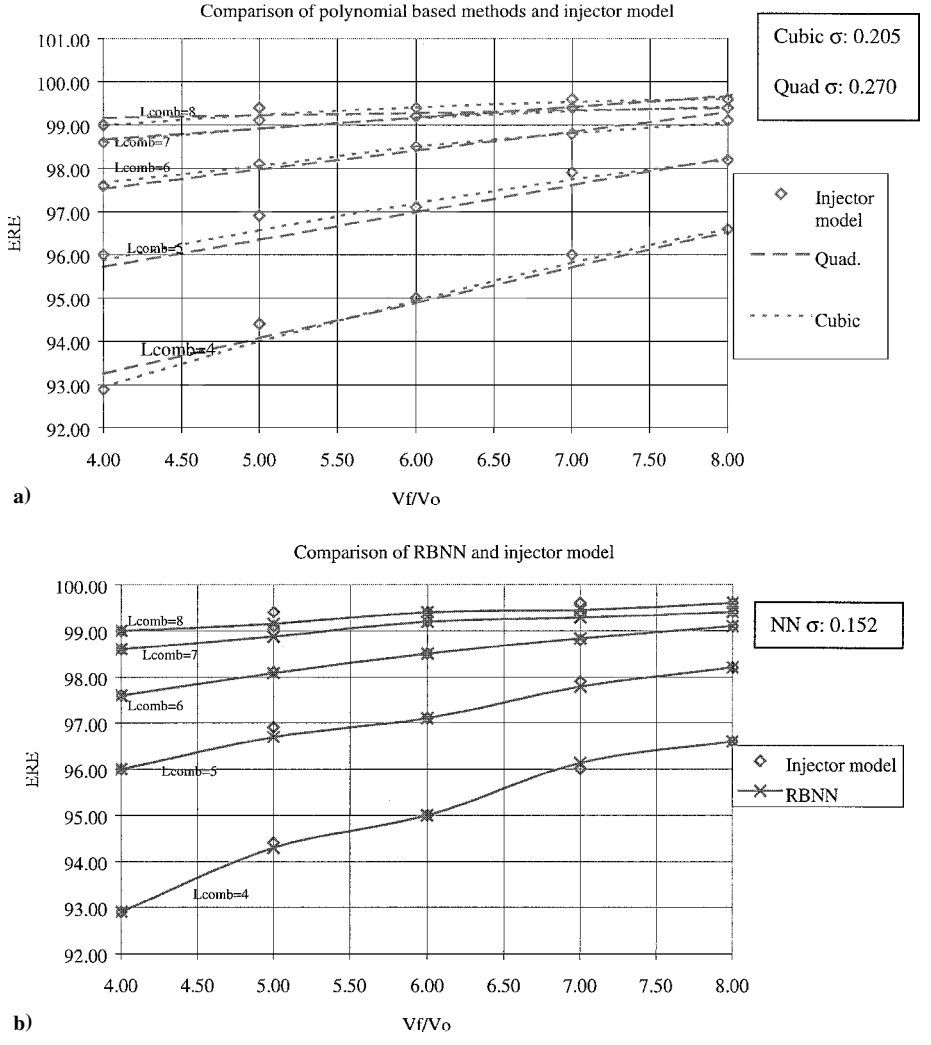

Fig. 4 Assessment of predictive capabilities of a) polynomial based method and b) RBNN for *ERE*, 15 training/mapping points 10 test points.

C. RBNN-Enhanced Response Surface

It has been demonstrated that the RBNN can be used to generate confidently additional design points. Additional design points generated by the RBNN are added to the original data set to form the enhanced data set. This enhanced data set is used for further analysis to evaluate the performance of the RSM with the larger number of design points. The enhanced data set for *ERE* has 15 points from the injector model and 10 from the RBNN, for a total of 25 points. The enhanced data set for Q has 9 points from the injector model and 4 from the RBNN, for a total of 13 points. The entire optimization analysis was redone with the enhanced data set. On this enhanced data set, the full quadratic response surface seems already appropriately constructed, and invoking the statistical analysis generates no reduced model. With the added data in the enhanced data set, it is now possible to obtain a fit for *ERE* that is fourth order in V_f/V_0 and fourth order in L_{comb}. Q can now be fit with a cubic in V_f/V_0 and a quadratic in O/F. This is now possible because a combination of three different values of O/F, five different values of V_f/V_0, and five different values of L_{comb} are available. The cubic fit for *ERE* and Q obtained from JMP are

$$\begin{aligned} ERE = {} & 48.813 + 4.807V_f/V_0 + 14.274L_{\text{comb}} - 0.141(V_f/V_0)^2 \\ & - 0.930(V_f/V_0)L_{\text{comb}} - 1.352(L_{\text{comb}})^2 \\ & + 0.0003(V_f/V_0)^3 + 0.0154(V_f/V_0)^2 L_{\text{comb}} \\ & + 0.0463V_f/V_0(L_{\text{comb}})^2 + 0.042(L_{\text{comb}})^3 \end{aligned} \tag{12}$$

and

$$\begin{aligned} Q = {} & -0.301 + 0.323O/F + 0.285V_f/V_0 - 0.0189(O/F)^2 \\ & - 0.094(O/F)V_f/V_0 + 0.00475(V_f/V_0)^2 \\ & + 0.00474(O/F)^2 V_f/V_0 + 0.0008(O/F)(V_f/V_0)^2 \\ & + 0.000058(V_f/V_0)^3 \end{aligned} \tag{13}$$

The quadratic fits from JMP are

$$\begin{aligned} ERE = {} & 69.68 + 2.088V_f/V_0 + 6.024L_{\text{comb}} - 0.042(V_f/V_0)^2 \\ & - 0.190(V_f/V_0)L_{\text{comb}} - 0.139(L_{\text{comb}})^2 \end{aligned} \tag{14}$$

and

$$\begin{aligned} Q = {} & 0.812 - 0.045O/F + 0.067V_f/V_0 + 0.0097(O/F)^2 \\ & - 0.028(O/F)V_f/V_0 + 0.0105(V_f/V_0)^2 \end{aligned} \tag{15}$$

1. *Comparison of Fits with the Original Response Surfaces*

Comparison of the enhanced response surfaces with the original response surfaces indicates that the extra data produced with the RBNN generally improve the quality of the curve fit. The adjusted rms error for *ERE* on the original set is 0.211 and 0.083 for quadratic and cubic fits, respectively. On the enhanced data set, it is 0.179 and 0.100 for the quadratic and cubic fits, respectively. The slight increase in the error in the case of the cubic fit may be due to

Table 9 Optimum values obtained with cubic and quadratic for case 1 (enhanced data set)[a]

		Cubic					Quadratic				
W_{ERE}, (s)	W_Q, (t)	O/F	V_f/V_0	L_{comb}, in.	*ERE*, %	Q, Btu/in.^{2}-s	O/F	V_f/V_0	L_{comb}, in.	*ERE*, %	Q, Btu/in.^{2}-s
1	10	6.0	**5.54**	7.0	99.02 (98.90)	0.654 (0.658)	6.0	**5.01**	7.0	98.96 (98.70)	0.644 (0.664)
1	1	6.0	6.00	7.0	99.12	0.658	6.0	6.00	7.0	99.25	0.658
10	1	6.0	6.00	7.0	99.12	0.658	6.0	6.00	7.0	99.25	0.658

[a]Constraints: $4 \le O/F \le 6$, $4 \le V_f/V_0 \le 6$, and L_{comb} (in.) ≤ 7. Compare with Table 5.

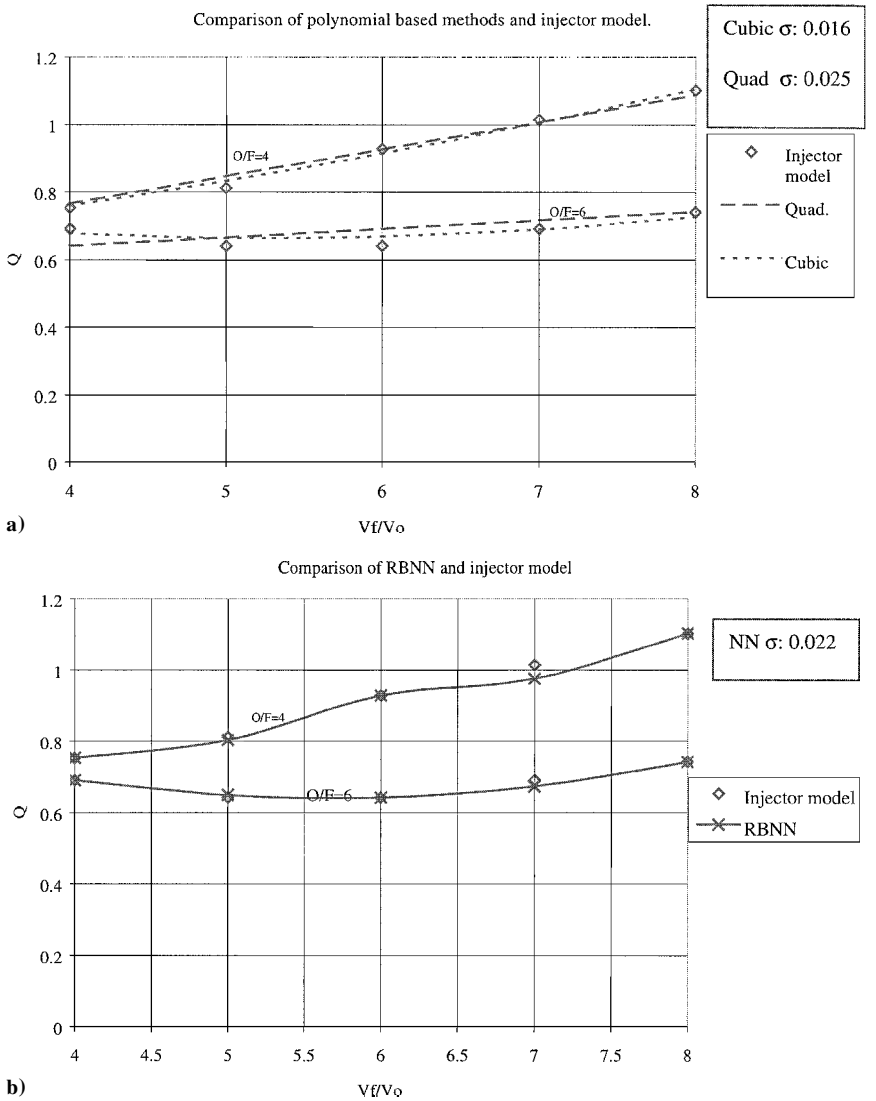

Fig. 5 Assessment of predictive capabilities of a) polynomial based methods and b) RBNN for Q, 9 training/mapping points, 4 test points.

noise related to the oversensitivity of the polynomial. However, this phenomenon may reflect that the level of the rms is low enough in either case so that no further improvement is accomplished. The adjusted rms error for Q with the original set is 0.039 and 0.040 for the quadratic and cubic fits, respectively. On the enhanced set it was 0.027 and 0.026 for the quadratic and cubic, respectively. With the exception of the cubic fit for *ERE*, the fits from the enhanced surface are improved over those from the original surface. Also, when optimum design points are examined, there is less difference between the quadratic and cubic fits on the enhanced surfaces than there is on the original surfaces.

2. *Comparison of Optimal Design Points*

The analysis for the three cases of optimization over the same three ranges of independent variables has been redone. The results of the optimization on surfaces generated from the enhanced data set are given in Tables 9–11. The predicted optimal design points using cubic and quadratic fits are generally close to each other. They are closer to each other on the reduced data set than on the surfaces generated using the original data set. One case where the cubic and quadratic optimum points are somewhat different is analyzed further. The results shown in Fig. 6 confirm the optimum value of velocity ratio on the quadratic fit to be lower than the cubic fit in this case. The enhanced set includes model data and RBNN predicted data. Quadratic and cubic are predicted by RSM. Given the weightings of 1.0 for *ERE* and 10.0 for Q, the optimizer has selects the minimum of Q for both fits. Because the curves exhibit different minimum points, the weightings force the selection of different optimum points. As already discussed, for the polynomial fits on the RBNN-enhanced data sets, the error of both quadratic and cubic

Table 10 Optimum values obtained with cubic and quadratic for case 2 (enhanced data set)[a]

		Cubic					Quadratic				
W_{ERE}, (s)	W_Q, (t)	O/F	V_f/V_0	L_{comb}, in.	*ERE*, %	*Q*, Btu/in.^{2}-s	O/F	V_f/V_0	L_{comb}, in.	*ERE*, %	*Q*, Btu/in.^{2}-s
1	10	6.0	**5.54**	7.0	99.02 (98.90)	0.654 (0.658)	6.0	**5.01**	7.0	98.96 (98.70)	0.644 (0.664)
1	1	6.0	**6.33**	7.0	99.18 (99.10)	0.663 (0.666)	6.0	**6.04**	7.0	99.26 (99.20)	0.659 (0.642)
10	1	6.0	7.00	7.0	99.30	0.681	6.0	7.00	7.0	99.46	0.693

[a]Constraints: $4 \le O/F \le 6$, $5 \le V_f/V_0 \le 7$, and L_{comb} (in.) ≤ 7. Compare with Table 6.

Table 11 Optimum values obtained with cubic and quadratic for case 3 (enhanced data set)[a]

		Cubic					Quadratic				
W_{ERE}, (s)	W_Q, (t)	O/F	V_f/V_0	L_{comb}, in.	*ERE*, %	*Q*, Btu/in.^{2}-s	O/F	V_f/V_0	L_{comb}, in.	*ERE*, %	*Q*, Btu/in.^{2}-s
1	10	6.0	6.00	7.0	99.12	0.658	6.0	6.00	7.0	99.25	0.658
1	1	6.0	**6.33**	7.0	99.19	0.663	6.0	**6.04**	7.0	99.26	0.659
10	1	6.0	8.00	7.0	99.42	0.725	6.0	7.95	7.0	99.57	0.746

[a]Constraints: $4 \le O/F \le 6$, $6 \le V_f/V_0 \le 8$, and L_{comb} (in.) ≤ 7. Compare with Table 7.

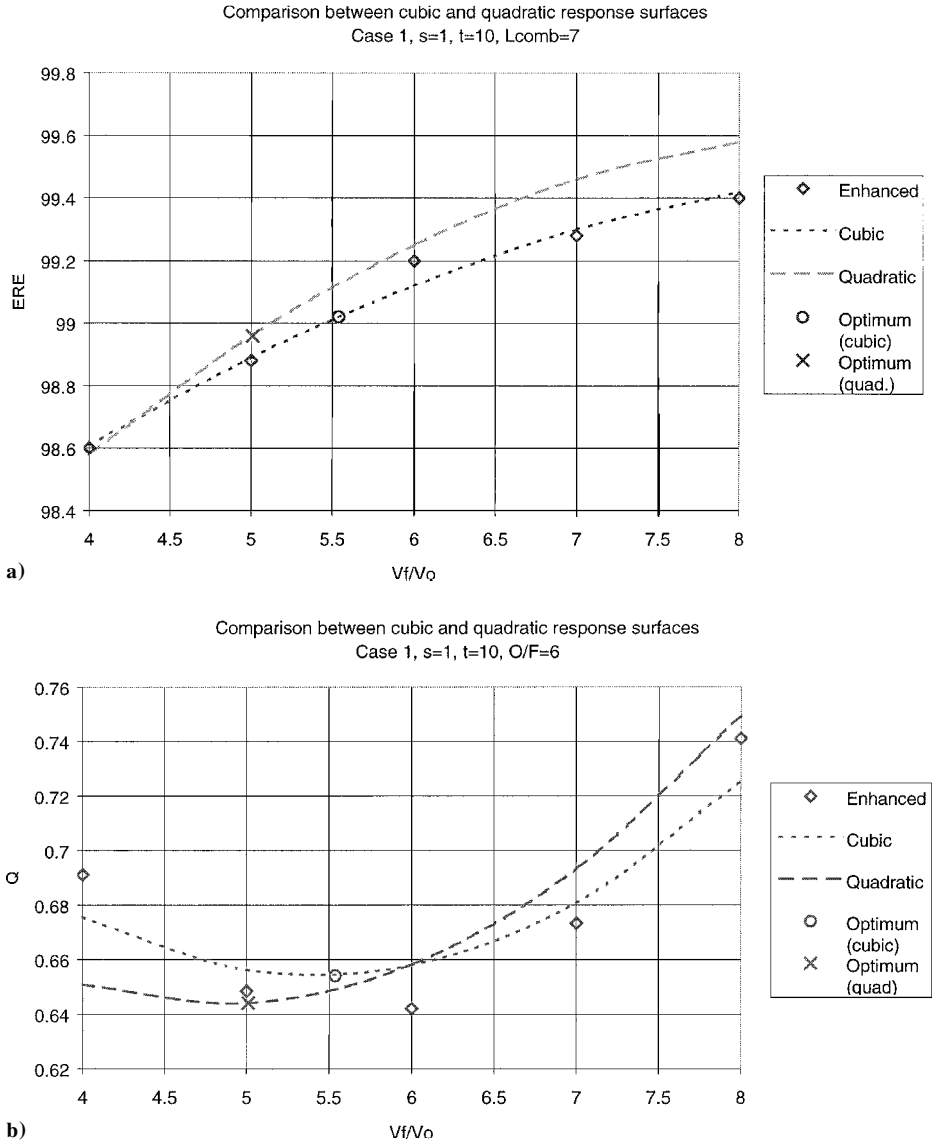

Fig. 6 Assessment of performance of cubic and quadratic response surfaces for case 1 of a) *ERE*, 25 training/mapping points and b) *Q*, 13 training/mapping points.

polynomials are more comparable than in the original analysis. At the upper limit of available data for combustor length, 8 in., the *ERE* curves tend to flatten out. This causes some difficulty in locating the optimum and may cause more noticeable differences between the different polynomials. However, different optimal designs selected by different polynomials under such a circumstance are not important because these yield very similar injector performance.

V. Summary and Conclusions

RSM and NN techniques have been applied to the optimization of a simple rocket injector. Injector performance, as measured by *ERE* and chamber wall heat flux, has been modeled as a function of propellant velocity ratio, oxidizer to fuel ratio, and combustor length. An empirical injector model was used to calculate 45 design points in the design space. The responses in these original data were fit to the input variables using both quadratic and cubic polynomials using RSM as embodied in the JMP software package. The fits were evaluated relative to each other using the adjusted rms error and the ability of the fit to predict additional data from the empirical model. Optimization studies were conducted using each fit over three ranges of independent variables for different weightings, or desirability, of the responses. Optimum points were located for each of 18 combinations of data range, curve fit, and response weighting using Microsoft Excel-SOLVER. These optimum points were then compared to exact values calculated from the empirical model.

RBNN were trained on the original data set using functions from the MATLAB-NN toolbox. Issues relevant to obtaining satisfactory RBNN performance and to enhancing the RSM for the current problem were investigated. An RBNN was trained on the original data set. The RBNN was compared to the RSM in terms of ability to predict additional data in the design space. The RBNN was then used to generate new data, which were combined with the original data set to form an enhanced data set. The optimization procedure was repeated using the enhanced data set. Quality of fit and location of optimal points were used to compare the fits from the enhanced surface with those from the original data set.

Based on the effort described, the following observations are made for the present injector design system.

1) The cubic fit was superior to the quadratic fit on the modest-sized original data set by each measure investigated: first, in terms of adjusted rms error, second, in that the optimal design points were closer to the data from the empirical model, and finally, in terms of the rms error relative to predicting additional data in the design space.

2) There was not a significant difference in the performance of Solverbe and Solverb in terms of generating the RBNN on the original data set.

3) The RBNN was able to generate additional design data for ERE with better accuracy than either the quadratic or cubic fit. For Q, the RSM error from both polynomials on the original data set and for the RBNN were all very low. The rms error for the RBNN fell between that of the two polynomials.

4) Comparison of the quality of polynomial fits on the original and RBNN-enhanced data sets indicated generally better fits on the enhanced surface. The only exception was that the cubic fit for *ERE* was slightly poorer on the enhanced surface. However, the error was already small.

5) The optimal points located on the quadratic and cubic surfaces generated from the enhanced data set were, for the same case, consistently closer to each other than were the fits from the original data set. Also, the original data set cubic fit was closer to the enhanced data set cubic fit than were the quadratic fits for the two data sets.

The preceding observations, taken together, indicate that RSM, when used in conjunction with NN, is capable of producing meaningful optimization studies with modest amounts of data. NN can be used to produce data of sufficient accuracy to actually improve the quality of polynomial fit in the RSM. Because accurate data (either from physical tests or CFD analysis) are time consuming and expensive to obtain for rocket engine injectors, the technique of coupling RSM with NN holds significant potential for their optimization.

Acknowledgment

The present work has been supported in part by NASA Marshall Space Flight Center.

References

[1]Rupe, J. H., "An Experimental Correlation of the Nonreactive Properties of Injection Schemes and Combustion Effects in a Liquid Rocket Engine," NASA TR 32-255, 1965.

[2]Pieper, J. L., "Oxygen/Hydrocarbon Injector Characterization," U.S. Air Force Phillips Lab., PL-TR 91-3029, Oct. 1991.

[3]Nurick, J. H., "DROPMIX-A PC Based Program for Rocket Engine Injector Design," *JANNAF Propulsion Conference*, WY, Nov. 1990.

[4]Dickerson, R., Tate, K., and Nurick, W., "Correlation of Spray Injector Parameters with Rocket Engine Performance," U.S. Air Force Rocket Propulsion Lab., AFRPL-TR-68-11, Jan. 1968.

[5]Pavli, A. L., "Design and Evaluation of High Performance Rocket Engine Injectors for use with Hydrocarbon Fuels," NASA TM 79319, 1979.

[6]Tucker, P. K., Shyy, W., and Sloan, J. G., "An Integrated Design/Optimization Methodology for Rocket Engine Injectors," AIAA Paper 98-3513, July 1998.

[7]Calhoon, D., Ito, J., and Kors, D., "Investigation of Gaseous Propellant Combustion and Associated Injector-Chamber Design Guidelines," Aerojet Liquid Rocket Co., NASA CR-121234, Contract NAS3-13379, July 1973.

[8]Myers, R. H., and Montgomery, D. C., *Response Surface Methodology—Process and Product Optimization Using Designed Experiment*, Wiley New York, 1995, pp. 16–67, 208–265.

[9]JMP, Ver. 3, SAS Inst., Inc., Cary, NC, 1995.

[10]Microsoft Excel 97, Microsoft Corp., WA, Copyright 1985–1996.

[11]Lasdon, L. S., Waren, A., Jain, A., and Ratner, M., "Design and Testing of a Generalized Reduced Gradient Code for Nonlinear Programming," *ACM Transactions on Mathematical Software*, Vol. 4, No. 1, 1978, pp. 34–50.

[12]Hertz, J., Krogh, A., and Palmer, R. G., *Introduction to the Theory of Neural Computation*, Lecture Notes, Sante Fe Inst. Studies in the Sciences of Complexity, Addison Wesley Longman, Reading, MA, 1991, pp. 89–156.

[13]Vaidyanathan, R., Papila, N., Shyy, W., Tucker, P. K., Griffin, L. W., Haftka, R. T., and Fitz-Coy, N., "Neural Network and Response Surface Methodology for Rocket Engine Component Optimization," AIAA Paper 2000-4880, Sept. 2000.

[14]Dernuth, H., and Beale M., MATLAB Neural Network Toolbox, Mathworks Natick, MA, 1992.

[15]Papila, N., Shyy, W., Fitz-Coy, N., and Haftka, R. T., "Assessment of Neural Net and Polynomial-Based Techniques for Aerodynamic Applications," AIAA Paper 99-3167, June–July 1999.