Optimally-Tuned Cascaded PID Control using Radial Basis Function Neural Network Metamodeling

H. Wahid[1, 2], Q.P. Ha[1]
[1]Faculty of Engineering and Information Technology
University of Technology Sydney (UTS)
Broadway, NSW 2007, Australia.
hbinwahi@eng.uts.edu.au, quangha@eng.uts.edu.au

M.S. Mohamed Ali[2]
[2]Faculty of Electrical Engineering
Universiti Teknologi Malaysia (UTM)
Skudai 81310, Johor Bahru, Malaysia.
sultan_ali@fke.utm.my

Abstract

Dynamic systems are quite often non-linear and require a complex mathematical model. For their optimal control, it has been always a requirement to tune the controller parameters to achieve the best performance. Parameter tuning in complex systems is predominantly a time-consuming task, even with high performance computers. This paper provides an overview of metamodeling and demonstrates how it can be applied to efficiently tune the control parameters of a typically nonlinear and unstable process, the ball and beam system. Here, the metamodel is realized with a radial basis function (RBF) neural network to derive the PID parameters subject to an optimal criterion. The proposed approach is benchmarked with a commonly-used tuning technique.

1. Introduction

In the control design for dynamic systems, one has often a complicated mathematical model, obtained from fundamental laws of physics, chemistry, or economics, for instance, that govern dynamics of the concerning system. To achieve the control objective, it is normally desired to find the optimal parameters for a controller that would give a highest performance index to the system outputs.

The optimisation algorithms might be too computationally expensive owing to the complexity of the actual model. The computational time required to simulate and use the actual model for further processing might be very long and thus it becomes impractical to rely exclusively on time-consuming simulation results for a too complicated model. Thus there is a need for metamodeling, that is, for the determination of simpler models that involve less computation but represent adequately a good approximation for the system dynamical behaviour.

In this work, a neural network-based metamodel is applied to optimise the controller parameters for a typically nonlinear and unstable process, the ball and beam system. The remainder of this paper is organised as follows. Section 2 gives an overview of the current metamodeling techniques. A rationale for tuning the PID gains using the metamodeling approach is provided in Section 3. The proposed technique, featuring a radial basis function with a target data set obtained from minimization of an integral performance index, is described in Section 4. The ball and beam system control using the approach is discussed in Section 5. Section 6 shows the simulation results, followed by a comparison analysis. Finally, a conclusion is drawn in Section 7.

2. Metamodeling overview

Metamodeling, considered generally as an *explicit* description of how a domain-specific model is built for a complex system, has been successfully used in many fields where complicated computer models of an actual system exist but they may require a considerable amount of running time. Models involving finite element and fluid dynamics analysis or multi-objective optimisation algorithms with many parameters are some typical examples.

Metamodeling evolves from the classical Design of Experiments (DOE) theory, in which polynomial functions are used as response surfaces, or metamodels. Nowadays there exist a number of metamodeling techniques, such as neural networks [1], Kriging [2], Radial Basis Functions (RBF) [3], Multivariate Adaptive Regression Splines (MARS) [4], Least Interpolating Polynomials [5] etc. Nevertheless, there is no conclusion about which model is definitely superior to the others. However, insights have been gained through a number of recent studies, whereby Kriging models, Gaussian and RBF processes are intensively investigated [6-8].

In general the Kriging model is more accurate for nonlinear problems and also flexible in either interpolating the sample points or filtering noisy data. However, it is difficult to obtain and use because a global optimization process is applied to identify the maximum likelihood estimators. On the contrary, a polynomial model is easy to construct, clear on parameter sensitivity, and cheap to work with but is less accurate than the Kriging model [7].

The RBF model, especially the multi-quadric RBF, can interpolate sample points and at the same time is easy to construct. It thus seems to reach a trade-off between Kriging and polynomials. Recently, a new model called Support Vector Regression (SVR) was used and tested [8]. SVR achieved high accuracy over all other Metamodeling techniques including Kriging, polynomial, MARS, and RBF over a large number of test problems. It is still unclear, however, what are the fundamental reasons for the SVR outperformance over others.

3. PID tuning and cascade control

The PID algorithm has been the most popular feedback controller used within process control over the years. It is a well understood algorithm that can provide excellent control performance despite the varied dynamic characteristics of process plant. Even though PID controllers are commonly used, they are not always used in the best way because the controller may not be well adjusted. It is difficult to tune three parameters by trial and error, especially when several PID controllers exist in the loop in a multi-input multi-output (MIMO) system.

There are many tuning techniques for PID control. They methods can be classified as: i) empirical methods such as the Ziegler-Nichols (ZN) method and the Internal Model Control (IMC) [9], ii) analytical methods such as root locus based techniques [9], iii) methods based on optimisation such as the iterative feedback tuning (IFT) [10] and genetic algorithm tuning technique [11]. The automatic tuning software for PID is also being studied recently. For example, a key method for auto-tuning was introduced by Aström and Hagglund [12] using the relay feedback concept. As the relay experiment gives the ultimate period T_u and ultimate gain K_u, a reasonable PID controller based on this point may then be calculated. In general, these techniques work well for a basic control feedback configuration, but they are not optimal for systems with multiple controllers and with nonlinear characteristics.

In this paper a metamodeling technique utilising the radial basis functions is proposed as an alternative to tune the PID gains subject to an optimal criterion. Unlike the above mentioned techniques, this method can handle the tuning task for multiple controllers, regardless of the complexity, the order and nonlinear characteristics of the system. In addition, with the availability of a simpler model, several design issues such as what-if analysis, prediction of a system output, optimisation and verification and validation of simulation models can be done in a computationally-efficient way since computing the output of an optimised metamodel (say a neural network for example) will be just in a matter of minutes. For comparison example, a finite element simulation program solving a microwave passive/active circuit problem took about 8 hours execution time on a Pentium PC, as reported by Tsai *et al*. [13].

In this investigation, a cascaded PID control configuration will be used to yield better dynamic performance. Using cascade control, there are two PID controllers arranged with the outer loop PID providing the set point of the inner loop. The outer loop controller controls the primary physical variable while the inner loop, which reads the output of outer loop controller as its set point, usually controls a more rapid changing variable corresponding to the smallest time-constant uncompensated dynamics. A cascade sliding mode-PID control scheme was proposed in [14], where the main concern is suppressing overshoot. In this work, the same principle is applied using a metamodel subject to an optimal criterion.

4. Radial Basis Function model

In this work, a Radial Basis Function Neural Network (RBFNN) is used as the proposed metamodel to approximate the mapping of the controller gains and an optimal performance index function.

The radial basis functions were first used to design Artificial Neural Networks in 1988 by Broomhead and Lowe [15]. The network consists of three layers: an input layer, a hidden layer and an output layer. Here, R denotes the number of inputs while Q the number of outputs. The output of the RBF NN, e.g. for $Q = 1$, is calculated as

$$\eta^1(x,w) = \sum_{k=1}^{S1} w_{1k}\phi\left(\left\|x - c_k\right\|_2\right), \qquad (1)$$

where $x \in \Re^{R\times 1}$ is an input vector, $\phi(\cdot)$ is a basis function, $\|\cdot\|_2$ denotes the Euclidean norm, w_{1k} are the weights in the output layer, $S1$ is the number of neurons (and centres) in the hidden layer and $c_k \in \Re^{R\times 1}$ are the RBF centres in the input vector space. Equation (1) can also be written as,

$$\eta(x,w) = \phi^T(x)w \ , \qquad (2)$$

where

$$\phi^T(x) = \left[\phi_1\left(\left\|x - c_1\right\|\right) \ \cdots \ \cdots \ \phi_{S1}\left(\left\|x - c_{S1}\right\|\right)\right] \qquad (3)$$

and

$$w^T = \left[w_{11} \quad w_{12} \quad \cdots \quad w_{1S1}\right] . \qquad (4)$$

The output of the neuron in a hidden layer is a nonlinear function of the distance given by:

$$\phi(x) = e^{-\frac{x^2}{\beta^2}}, \qquad (5)$$

where β is the spread parameter of the RBF. For training, the least squares formula was used to find the second layer weights while the centers are set using the available data samples.

The radial basis function neural network offers several advantages compared to the Multilayer Perceptrons. RBFNNs have also been successfully used in engineering design, see e.g. [16], and manufacturing, see e.g. [17]. These advantages include the ability to effectively generate multidimensional interpolative approximations, to yield robustness and reliability in a computationally-aided design.

To incorporate the cascade control principle, here the radial basis function metamodel is used to obtain minimum of the RBFNN output, which is the approximation of an optimal performance index, the *integral of time multiplied by the absolute value of error* (ITAE) given by:

$$\text{ITAE} = \int_0^{\infty} t\left|\left(y_d(t) - y(t)\right)\right| dt, \qquad (6)$$

where y_d is the desired output (set point) while y is the actual output of the control system. This criterion is chosen as it will produce smaller overshoots and oscillations than the others criteria such as *integral of absolute magnitude of error* (IAE) and *integral of the square of error* (ISE). The input vector contains the set of PID controller parameters K_{p1}, K_{i1} and K_{d1} for the outer loop (displacement controller) and K_{p2}, K_{i2} and K_{d2} for the inner loop (motor gear angle controller). The target class vector is obtained from the computed performance index (6) for the cascade control closed-loop system. The following algorithm is proposed:

1. Define the input design space, *D*, which consists of a set of initial values of the controller parameters.
2. Obtain the ITAE for the control outer loop. Create the target data set, *T*, which consists of the normalized ITAE.
3. Fit the RBFNN using *D* and *T*.
4. Evaluate the RBFNN on a denser input space, *D'*.
5. Find the minimum of the RBFNN output (estimated $\bar{E}$). The corresponding controller gains that minimized the RBF output will be the gains to be verified in actual model simulation.
6. Repeat step 1 to 6 until reach the goal, i.e., the mean squared error of RBFNN smaller than a prescribed threshold.

5. Case study: A Ball and Beam system

The ball and beam system is a nonlinear and unstable system, thus providing a challenge to the control engineers or researchers. Basically, there are two types of configuration. The first configuration normally called 'Ball and Beam Balancer', which the beam is supported in the middle, and rotates against its central axis. Most ball and beam systems use this type of configuration, see, e.g., Rosales *et al.* (2004) [18]. The ball and beam balancer is easy to build and its mathematical model is relatively simple. In the second configuration called 'Ball and Beam Module', the beam is supported on the both sides by two level arms. One of level arms acts as the pivot and the other is coupled to motor output gear. Although having a more complicated mathematical model, the ball and beam module system has an advantage in that relatively small motors can be used in couple with a gear box [19]. This type of configuration will be used in our work.

Here, the aim is to control the rolling ball at the desired position through its acceleration. Thus, it will imply the presence of two integrators in series with the dynamics of the beam, which result in an open-loop unstable and non-linear system [20]. For this a number of controllers have been proposed, including nonlinear control [21]. A linearised feedback technique can be applied, see e.g., [22], from which the stability analysis can be obtained, based on linear state-space model or transfer function.

As shown in Fig. 1, a ball is placed on a beam where it is free to roll at horizontal axis along the length of the beam. A lever arm is attached to the beam at one end and a servo gear at the other. The servo gear turns by an angle θ, and the lever changes the angle of the beam by α. The force that accelerates the ball as it rolls on the beam come from the component of gravity that acts parallel to the beam. The ball actually accelerates along the beam by rolling, but we can simplify the derivation by assuming that the ball is sliding without friction along the beam. The mathematical modeling of ball and beam system consists of the ball on the beam dynamics, alpha-theta relation and DC servomotor model.

By using the Lagrange method or the free body for the acting forces shown in Fig. 1, the ball and beam dynamics can be obtained as in [23]. The beam angle (α) can be related with motor gear angle (θ) by approximate linear equation $\alpha L = \theta d$, where d is the lever arm offset and L is the beam length. The transfer function from the input voltage (V) to angle θ for the electromechanical system with a negligible armature inductance ($L_a \cong 0$) can be obtained as:

$$\frac{\theta(s)}{V_{in}(s)} = \frac{a_m}{s^2 + b_m s}, \qquad (7)$$

where $a_m = \dfrac{K_m K_g}{R_a J_1}$, $b_m = \dfrac{B}{J_1} + \dfrac{(K_m K_g)^2}{R_a J_1}$, R_a is the motor armature resistance, K_m is the back emf/torque constant, $K_g = \theta^{-1}\theta_m$ is the gear ratio, $J_1 = K_g{}^2 J_m + J_L$ is the total moment of inertia, and $B = K_g{}^2 B_m + B_L$ is the total damping. The linearised state-space equations for the open-loop system can be obtained as:

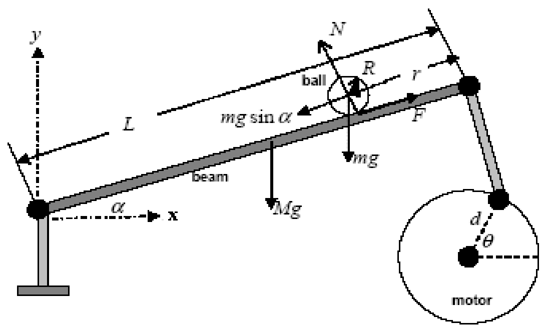

Figure 1. Force acting on the ball and beam system

$$\begin{bmatrix} \dot{r} \\ \ddot{r} \\ \dot{\theta} \\ \ddot{\theta} \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & -\dfrac{gd/L}{(1+J_b/mR^2)} & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & -b_m \end{bmatrix} \begin{bmatrix} r \\ \dot{r} \\ \theta \\ \dot{\theta} \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 0 \\ a_m \end{bmatrix} V_{in}, \quad (8)$$

representing a marginally-stable system that will be stabilised using the two PID controlled loops respectively for the ball displacement and for the beam angle, as shown in the block diagram of Fig. 2. In equation (8), J_b is the moment inertia of the ball, R is radius of the solid ball, $\ddot{r}$ is its acceleration, m is its mass, g is gravitational constant, α is the beam angle, and $\dot{\alpha}$ is its angular velocity.

6. Simulation and discussion

The following numerical parameters are used: $J_1 = 7.35\times10^{-4}\ kgm^2$, $B = 1.6\times10^{-3}\ Nms/rad$, $K_g = 75$, $R_a = 9\Omega$, $L_a = 0$, and $K_m = 0.0075Nm/A$. Moment of inertia of the solid ball given by $J_b = 2mR^2/5\ kgm^2$, where $m = 0.110\ kg$ and $R = 0.015\ m$.

The data sets *D* and *D'* are given in Table 1. The RBFNN used to fit the data set *D* is designed with 16 radial basis centres. These centres are added one by one until the RBFNN reaches a prescribed error goal, set at 0.01. In our design, the width of an area in the input space to which each neuron responds is determined by the spread β, which is selected as 150, sufficiently large to cover overlapping regions of the input space.

Table 1. Controller parameters used for simulation

Initial and Large Data Sets		
Initial Data Set D	K_{p1}	{1, 4, …, 10}
	K_{i1}	{0, 0.04,.., 0.1}
	K_{d1}	{0, 4,…, 10}
	K_{p2}	{10, 80, …, 200}
	K_{i2}	{0, 0.04, …, 0.1}
	K_{d2}	{0, 0.08,.., 0.2}
Total number of data configurations.		**729**
Large Data Set D'	K_{p1}	{1, 2, …, 10}
	K_{i1}	{0, 0.02,.., 0.1}
	K_{d1}	{0, 2,…, 10}
	K_{p2}	{10, 40, …, 200}
	K_{i2}	{0, 0.02, …, 0.1}
	K_{d2}	{0, 0.04,.., 0.2}
Total number of data configurations.		**32400**

Using MATLAB® on an INTEL® Core™ 2 Duo PC, it takes exactly **0.98 minutes** to complete steps 1-6 as explained in the previous section. The best gain that minimizes $\overline{E}$ is given in Table 2.

To verify the radial basis function metamodel, a SIMULINK block diagram was constructed as a target and evaluated for all the 32400 cases in *D'* using the same PC and the error, $\overline{E}$ was also computed. The simulation took 39.01 *minutes* to complete, giving the gain that minimizes $\overline{E}$ as tabulated (see Table 2). The response of the ball displacement based on these PID controller parameters are as plotted in Figure 3. The output load disturbance is also applied in the entire process for the proposed metamodel method as well as the target model as illustrated in Fig. 1, and the load disturbance response is depicted in Fig.3. As can be seen from Fig. 3, with the PID controller parameters, a good transient response is obtained at the output of the ball and beam system. The response follows the changes of the desired input either in positive or negative direction. When a disturbance input affects the ball position (e.g. user touches the ball), the feedback control is still able to stabilise the oscillation within a reasonable range.

Table 3 shows a comparison of the results obtained. From the results (Table 2 and 3), it can be observed that the metamodel managed to approximate the global minimum of the error curve fairly well. Although the minimum of the normalized error for the metamodel and the actual model is slightly different (0.017 vs 0.0162), the actual ball displacement error E_d, obtained by using the optimal parameters from the ITAE criterion (6) for the metamodel and the system block in diagram Figure 2, remains the same (see Table 3).

Given the quick computational time for the proposed radial basis function metamodel, about 1 minute, compared to the time required for simulating the process with all values in *D*' (about 40 minutes), the results obtained indicate clearly an improvement. This will become more significant for a larger input space *D* or for a more complicated problem. Hence, metamodeling provides the designer with a quick estimate for a set of good parameters to begin with.

To evaluate the performance of the proposed method, a comparison study is included here with results obtained from a PID auto tuning technique based on relay feedback. For tuning PID controller parameters, a number of approaches have been in use, similarly to the Ziegler-Nichols rules. Initiated by Aström and Hagglund [12], commercial automatic tuning PID controllers are nowadays available from most hardware manufacturers for the industrial usage.

In this analysis, the more advanced configuration for the relay feedback proposed in [23] will be used as a benchmark, where the tuning of the cascade controllers should be performed with the inner loop first and then the outer loop.

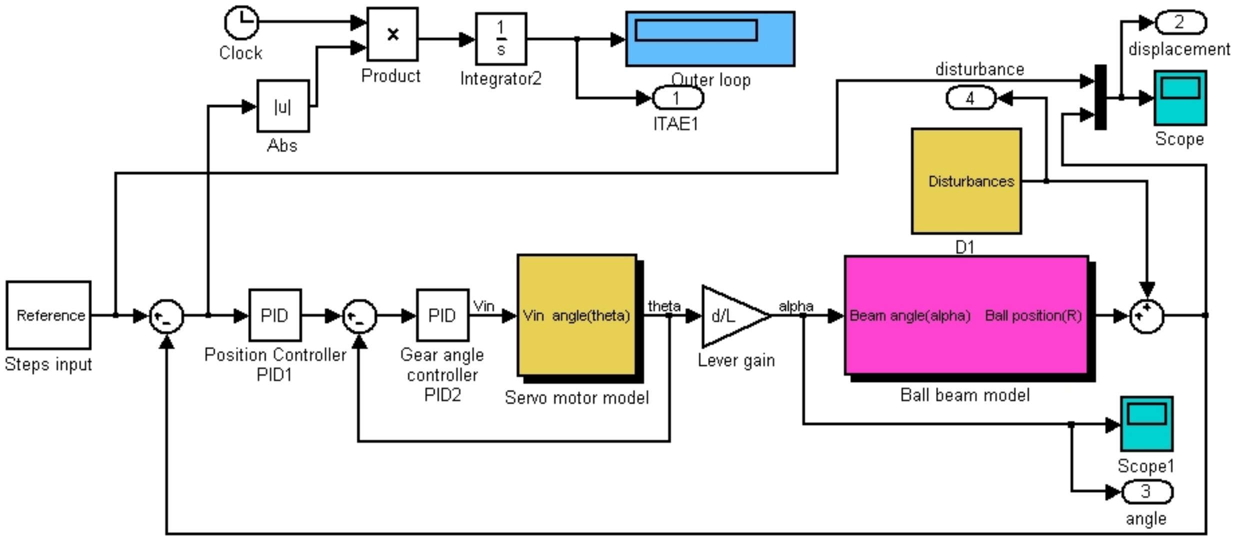

Figure 2. A Ball and Beam System as implemented in MATLAB® SIMULINK®

Table 2. Ball and Beam controller parameters used for simulation

Controller Parameter	RBFNN metamodel	Actual evaluation
K_{p1}	9	9
K_{i1}	0	0
K_{d1}	6	8
K_{p2}	90	170
K_{i2}	0.1	0.1
K_{d2}	0.2	0

Table 3. Global minimum for the metamodel and the actual model

	Metamodel	Actual
min($\overline{E}$)	0.0170	0.0162
Actual E_d at argmin($\overline{E}$)	0.4139	0.4139

Table 4. PID parameters with metamodel and relay feedback auto-tuning

Controller Parameter	RBFNN metamodel	Relay Auto-tuning
K_{p1}	9	25
K_{i1}	0	3
K_{d1}	6	26
K_{p2}	90	0.93
K_{i2}	0.1	0.95
K_{d2}	0.2	0.23

Table 5. Step response characteristics

Method	OS (%)	T_p (s)	T_r (s)	T_s (s)	e_{ss} (%)
Metamodel (small space)	10	2	1.0	4	0
Metamodel (larger space)	0	NA	1.5	3	0
Relay auto tuning	50	2.5	1.0	11	0.01

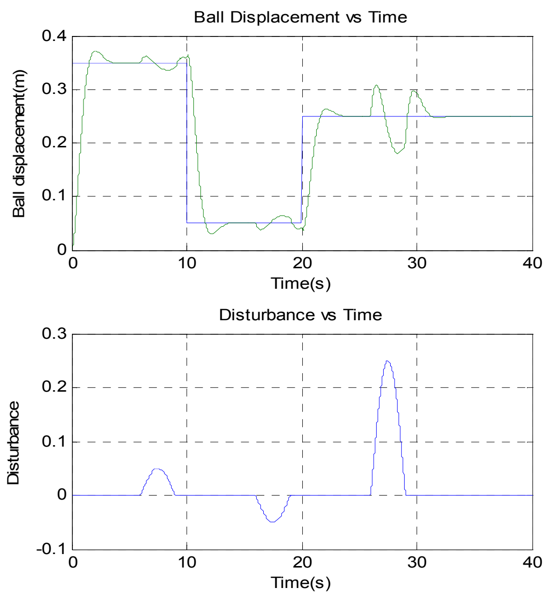

Figure 3. Displacement responses with disturbance.

The PID parameters for the proposed radial basis network metamodel and the relay feedback auto-tuning technique using Ziegler-Nichols rules are summarised in Table 4. From the table, we can observe that the PID parameters using the RBFNN metamodel method and relay feedback auto-tuning method are different although both methods are able to stabilise the unstable ball and beam system. For the outer loop, the proposed approach yields a PD controller instead of PID, implying that the integral action have to be minimized or omitted as possible. On the other hand, the relay auto-tuning method gives higher values of integral gain are obtained for both loops. As a consequence, the oscillatory response of the internal loop affects the tuning of PID parameters in the outer loop. The step responses obtained from using the metamodel and relay auto-tuning method are depicted in Figure 4, while performance indices such as peak time (T_p), rising time (T_r), settling time (T_s), and steady state error (e_{ss}) are listed in Table 5.

It can be seen that in general the metamodel method has better performance than the relay auto-tuning method if based on a conventional tuning basis.

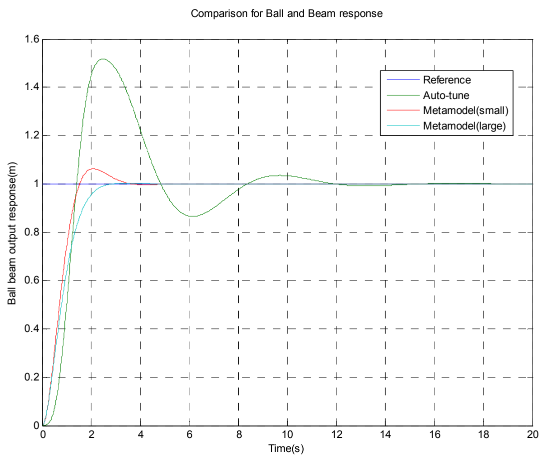

Figure 4. Performance comparison

7. Conclusion

This paper has presented the radial basis function metamodel method for cascade control of a complicated system involving multiple control loops. The effectiveness of the proposed approach is illustrated in tuning the PID parameters subject to an optimal integral criterion for a two-loop nonlinear unstable system, the ball and beam system. Comparing with PID tuning methods such as the commonly-used relay feedback auto-tuning, the proposed metamodel method gives good control performance. As to the computational efficiency, the computing time required to obtain a best fit set of the controller parameters is significantly fast compared to the time required for processing all possible values of the input space. Notably, this method is not only used for tuning control parameters, but also useful to approximate any complex relationships in systems with a large input space. For further improvement, it is envisaged that a more strategic data location will allow the creation of a more accurate metamodel using less data, and therefore, less time required to estimate the best parameter set.

Acknowledgments

This work is supported, in part, by and the Centre of Excellence programme, funded by the Australian Research Council (ARC) and the New South Wales State Government. The second author would like to gratefully acknowledge UTS research grant 2006000848.

References

[1] Cresssie, N., "Spatial Prediction and Ordinary Kriging," *Mathematical Geology*, Vol. 20, No. 4, pp. 405-421, 1988.

[2] Papadrakakis, M., Lagaros, M. and Tsompanakis, Y., "Structural Optimization Using Evolution Strategies and Neural Networks," *Computer Methods in Applied Mechanics and Engineering*, 156, No.1-4, 309-333, 1998.

[3] Dyn, N., Levin, D. and Rippa, S., "Numerical Procedures for Surface Fitting of Scattered Data by Radial Basis Functions," *SIAM Journal of Scientific and Statistical Computing*, 7(2), 639-659, 1986.

[4] Friedman, J. H., "Multivariate Adaptive Regressive Splines," *The Annals of Statistics*, 19, No.1, 1-67, 1991.

[5] De Boor, C. and Ron, A., "On Multivariate Polynomial Interpolation," *Constructive Approximation*, 6, 287-302, 1990.

[6] Varadarajan, S., Chen, W. and Pelka, C. J., "Robust Concept Exploration of Propulsion Systems with Enhanced Model Approximation Capabilities," *Engineering Optimization*, Vol. 32, No. 3, 309-334, 2000.

[7] Wang, L., Beeson, D., Akkaram, S. and Wiggs, G., "Gaussian Process Metamodels for Efficient Probabilistic Design in Complex Engineering Design Spaces," in: *ASME 2005 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, Long Beach, California USA, September 24-28, DETC2005-85406.

[8] Clarke, S. M., Griebsch, J. H. and Simpson, T. W., "Analysis of Support Vector Regression for Approximation of Complex Engineering Analyses," *Transactions of ASME, Journal of Mechanical Design*, 127(6), 1077-1087, 2005.

[9] K.J. Aström and T. Hagglund, *PID Controllers: Theory, Design, and Tuning*, The Inst., Systems, and Automation Society, 1995.

[10] H. Hjalmarsson, M. Gevers, S. Gunnarsson, and O. Lequin, "Iterative feedback tuning: Theory and applications", *IEEE Control Systems Magazine*, vol. 18, 26-41, 1998.

[11] J.M. Herrero, X. Blasco, M. Martinez, and J.V. Salcedo, "Optimal PID Tuning with Genetic Algorithms for Non-Linear Process Models", in *IFAC World Congress*, Barcelona, Spain, 2002.

[12] K.J. Aström and T. Hagglund. *Automatic Tuning of PID Controllers*. Instr. Society of America, 1988.

[13] H. P. Tsai, Y. Wang, and T. Itoh, "An Unconditionally Stable Extended (USE) Finite-Element Time-Domain Solution of Active Nonlinear Microwave Circuits Using Perfectly Matched Layers", *IEEE Transactions on Microwave Theory and Techniques* Vol. 50, no.10, 2226-2232, 2002.

[14] T.H. Tran, Q.P. Ha, and H. T. Nguyen, "Robust non-overshoot time responses using cascade sliding mode-PID control", *Journal of Advanced Computational Intelligence & Intelligent Informatics*, Vol. 11, No. 10, 1224-1231, 2007.

[15] D. S. Broomhead and D. Lowe, "Multivariable functional interpolation and adaptive networks". *Complex Systems*, *2*, pp. 321-355, 1998.

[16] A. A. Mullur and A. Messac, "Extended Radial Basis Functions: More Flexible and Effective Metamodeling," *AIAA Journal*, 43, No. 6, 1306-1315, 2005.

[17] T. Yang, H.-C. Lin, and M.-L. Chen, "Metamodeling approach in solving the machine parameters optimization problem using neural network and genetic algorithms: a case study," *Robotics and Computer-Integrated Manufact.*, Vol. 22, pp. 332-331, 2006.

[18] Evencio A. Rosales, Bennett T. Ito, Katie A. Lilienkamp, and Kent H. Lundberg, An Open-Ended Ball-Balancing Laboratory Project for Undergraduates, *Proc. 2004 American Control Conference* Boston. Massachusetts, 1314-1318, 2004.

[19] Wen Yu and Floriberto Ortiz, Stability analysis of PD regulation for ball and beam system, *Proceedings of the 2005 IEEE Conference on Control Applications Toronto, Canada*, pp. 517-522, August 28-31, 2005.

[20] Peter E. Wellstead. *Introduction to Physical System Modelling*. Academic Press Limited, pp. 221-227, 2000.

[21] R.M. Hirschorn, "Incremental Sliding Mode Control of the Ball and Beam," *IEEE Transactions on Automatic Control*, Vol. 47, pp. 1696-1700, 2002.

[22] J. Hauser, S. Sastry and P. Kokotovic, "Nonlinear control via approximate input-output linearization: ball and beam example," *IEEE Trans. on Automatic Control*, Vol. 37, pp. 392-398, 1992.

[23] C.C. Hang, A.P. Loh, and V.U. Vasnani, "Relay Feedback Auto-tuning of Cascade Controllers", *IEEE Transactions On Control Systems Technology*, Vol. 2, No. 1, pp. 42-45, 1994.