M. Peek and P. J. Antsaklis, "Parameter Learning for Performance Adaptation in Large Space Structures" (Invited), P roc. o f t he 4 th I EEE I nternational S ymposium o n I ntelligent C ontro l , Albany, NY, Sept. 25-27, 1989.

# Parameter Learning for Performance Adaptation in Large Space Structures

Mark D. Peek

Tellabs Research Center
3702 N. Main St, Bldg 2, Suite101
Mishawaka, IN 46545

Panos J. Antsaklis

Dept.of Electrical & Computer Engr.
University of Notre Dame
Notre Dame, IN 46556

## Abstract

A parameter learning method is introduced. It is then used to broaden the region of operability of the adaptive control system of a space structure. The learning system guides the selection of control parameters in a process leading to optimal system performance; the method is a form of learning by observation and discovery. It is applicable to any system where performance depends on a number of adjustable parameters. A mathematical model is not necessary as the learning system can be used whenever the performance can be measured via simulation or experiment.

## 1. Introduction

Large flexible space structures pose unique control problems because of the complexity of their dynamic behavior, the limited knowledge of the model, the time-varying elements and the uncertainty of the environment in the types of disturbances that will be encountered. While adaptive control has shown potential in effectively controlling such systems [14], and offers good disturbance rejection, the region of operability is defined by the adaptive controller, the parameters of which are typically decided based on convergence and stability analysis only. This may place severe limitations on the performance of the compensated system, here a space structure. Autonomous systems require a high degree of flexibility to adapt to situations which cannot be predicted. This requires the ability to adapt to significant changes affecting the region of operability. Adaptive behavior of this type is not offered by conventional adaptive control systems. Thus, this seems to be an area in which contributions made in the field of machine learning may be applied effectively.

The goal of the machine learning method proposed in this paper is to broaden the region of operability of the adaptive control system by allowing the controller parameters to adapt to different plant and environmental conditions. These conditions cause the nominal adaptive system to exceed the tolerances of its design. The learning method uses knowledge of incremental changes in the conditions to make intelligent decisions regarding the best set of controller parameters to use when changes occur. The buildup of knowledge in the learning system eliminates some of the uncertainty in the adaptive system, making the controller more robust.

The learning method proposed in this paper is applicable to any system where performance depends on a number of adjustable parameters. The mathematical relation between the performance and the parameters does not need to be known. Given particular values for the parameters, performance is evaluated via computer simulation or physical experiment; if the mathematical relation is known, the performance evaluation can be done directly. The learning system determines the next set of parameters in a process leading to an optimum performance. In effect, the learning system guides the selection of parameters for optimization; this procedure is a form of learning by observation and discovery.

In this paper, the learning system is applied to determine the best parameter values for an adaptive controller controlling a large space antenna. The antenna model is described in detail in [10,14,18].

Section II presents a brief overview of machine learning. Section III describes the general format of the parameter learning system and Section IV describes the particular antenna parameter learning system. Section V presents the results of the "transient regulation experiment". In this experiment, the learning system determines the best parameter values to use when given different disturbances. Finally, Section VI contains conclusions and offers suggestions regarding future research.

## 2. Machine Learning Methods

We are interested in the ability of man-made systems to learn from experience and, based on that experience, improve their performance. Thus, we start with a working definition of such learning. Learning is the process whereby a system can alter its actions to perform a task more effectively due to increases in knowledge related to the task. The actions that a system may take depend on the nature of the system. For example, a control system may change the type of controller used, or vary the parameters of the controller, after learning that the current controller does not perform satisfactorily within a changing environment. Similarly, a robot may need to change its visual representation of the surroundings after learning of new obstacles in the environment. The type of action taken by the machine is dependent upon the nature of the system and the type of learning system implemented.

The ability to learn entails such issues as knowledge acquisition, knowledge representation, and some level of inference capability. Learning, considered fundamental to intelligent behavior, has been the subject of research in the field of Machine Learning for over twenty years and has gained a renewed interest in the Artificial Intelligence community.

Machine learning can be classified in two ways [15]. The first classification emphasizes the underlying learning strategy, and considers the amount of reinforcement and the inference scheme used in the learning algorithm. The existing learning strategies, in order of increasing complexity of the inference capabilities are:

1. Rote or Program Learning
2. Learning from Instruction
3. Learning by Induction
4. Learning from Observation and Discovery

The second classification of machine learning depends on the method used for knowledge acquisition. How new knowledge is acquired by the machine directly affects the level of learning achieved. Knowledge acquisition can be approached by considering the system to belong to one or more of the following descriptive categories:

1. A Black Box
2. The Structural Description
3. An Evolutionary Process

It should be pointed out that the two classifications of machine learning presented are not independent of each other, nor is one method used exclusively in a learning algorithm. Instead, learning systems use a combination of learning strategies and knowledge acquisition methods depending on the goals of the system.

Learning strategies can be differentiated by the type of inference and reinforcement methods they use. Inference is the ability to draw new conclusions from given facts, while reinforcement, also called credit assignment, increases the probability that correct actions will be taken again when the same situation is encountered. By increasing the burden of inference on the student, less intervention is required of the teacher in providing reinforcement. Two extremes exist in this spectrum, totally supervised learning, as in rote learning, and unsupervised learning, as in learning by observation and discovery.

M. Peek and P. J. Antsaklis, "Parameter Learning for Performance Adaptation in Large Space Structures" (Invited), P roc. o f t he 4 th I EEE I nternational S ymposium o n I ntelligent C ontro l , Albany, NY, Sept. 25-27, 1989.

The other two methods, learning from instruction and learning by induction fall in between the first two.

The simplest learning strategy is Rote Learning. This strategy has the simplest inference method, since no attempt is made to infer relationships between what is already known and newly acquired knowledge. There is also no reinforcement in the learning strategy. Rote learning is strictly a memorization process, where a teacher dictates facts to a student, who stores and retrieves the data without knowing their significance.

Learning from Instruction is the next level in the learning strategy hierarchy. Learning from Instruction is similar to rote learning, but with some inference capability added. The strategy depends on interaction between a student and teacher, but the student has a greater responsibility in the inference process.

In Learning by Induction (also called learning by analogy, by simile or by example), the system attempts to find generalized patterns from situations supplied by a teacher. This method builds on the last level, learning from instruction, with the addition of the generalization procedure which examines the positive features of each case and determines a more general characteristic that relates to each situation.

Learning from Observation and Discovery (also called learning from experience) is considered the highest form of learning, since it requires no teacher. The learning algorithm performs the role of both teacher and student. This form of learning is similar to learning by induction with the fundamental difference that no teacher is available to supply new information or cases, nor provide positive and/or negative reinforcement. The learning algorithm itself must attempt to determine when it is right or wrong, and correct itself.

The ability to acquire new knowledge is fundamental to learning. Thus, the form of knowledge acquisition influences the capability of the learning algorithm. As mentioned previously, the three approaches to learning based on acquisition depend on how the system is viewed, the black box approach, the structural description approach, and the evolutionary process approach.

The black box approach is concerned only with the input/output relationship of the system. The parameter adjustment method and the classification method are the two common methods available to implement this type of learning algorithm. Parameter adjustment is a popular form of learning because of its simplicity. Typically, the method uses a weighting function where the weights are adjusted based on a correct or incorrect response. Feature values are added or subtracted from the weighting function if the output is too high or low during the previous iteration [4,15]. A second method of black box learning is the classification method. This method quantizes features into clusters or ranges, reducing the amount of data that must be dealt with. From the quantized ranges, rules can be developed that relate specific actions to the occurrence of a specific quantized feature.

The Structural Description Approach is based on learning descriptive relationships between objects as well as features of the individual objects. The implementation is more complex than the Black Box approach because of the requirement for a descriptive language and the ability to have a dynamic rulebase.

The third method of knowledge acquisition is the Evolutionary process approach. The method, rooted in Darwin's theory of evolution, or survival of the fittest, is applied to a population of structures rather than organisms. Procedures have been developed to simulate biological reproduction, including crossover, mutation, and inversion.

Machine learning is currently being applied in control systems to enhance plant modeling, select control parameters and determine relationships between environmental effects and control system parameters [2,16]. Applied in these areas, machine learning is reducing modeling and environmental uncertainty and thus minimizing control energy. Alternate learning vehicles, in the form of expert system technology, and rule-based programming in particular, are providing new methods to implement learning strategies. Examples of machine learning in control include the inverted pendulum problem [1], a face milling control system [7], rule-based learning for fault tolerant flight control [13] and a genetic learning algorithm for the control of a gas pipeline system [12].

## 3 The Parameter Learning System (PLS)

A new learning method is introduced here for parameter learning. The main objectives in developing this method have been the effective use of all available information and its speed of response. This task appears plausible because the interest is in developing a learning method for a rather specific class of problems. Thus, the available information is rather well defined. It should be stressed that the more the system knows, the faster it can learn.

The role of the parameter learning system is to determine the best set of parameters given changing conditions in the target system's environment. Whether the parameter learning system is invoked depends upon the time restrictions placed on determining a new parameter set and whether learning is necessary. This can be seen in the functional diagram for parameter setting given in Figure 1. When the environmental conditions have been identified, the dictionary containing information about such conditions is consulted to determine if they are known by the system. If these conditions are known, the parameters can be set appropriately, and learning is not required. If the conditions are unknown, a decision to enable learning must be made. If time allows, learning can be enabled. Otherwise, the parameter values can be estimated from known conditions or may be left unchanged.

The functional diagram of the parameter learning system is given in Figure 2. The parameter learning system estimates an initial parameter set using the information from the dictionary or by some other method. This current set of parameter values, $X_k$, is fed back to the target system and the performance of the system is evaluated by computer simulation or physical experimentation. The parameter set, $X_k$, and the performance, J, are related by

$$J = f(X_k) \tag{3.1}$$

where the function $f(\cdot)$ is typically unknown. The performance of the system is evaluated using measurable quantities and is expressed as

$$J = g(Y_k) \tag{3.2}$$

where $g(\cdot)$ is a known function of the measurable quantities $Y_k$. As $X_k$ varies, the measurable quantities, $Y_k$, reflect the changes in system performance. J is then evaluated from $Y_k$ via (3.2). The performance of the system is then judged to be adequate or inadequate. If inadequate, a new parameter set $X_{k+1}$ is generated using some search algorithm. However, if the function $f(\cdot)$ is known, other methods may be used to generate $X_{k+1}$. This process continues until the performance is judged adequate. At that time, the best parameters found are stored in the dictionary for the given conditions and control is returned to the target system.

## 4 The Antenna Parameter Learning System (APLS)

In the antenna parameter learning system (APLS), the input to the target system is a disturbance. The search procedure used to generate a new parameter set $X_{k+1}$ is a modified version of the Hooke and Jeeves multidimensional search algorithm [6]. The estimation procedure is a grid search, or an interpolation routine using the information in the dictionary, if present. The performance of the system is evaluated using measurable system quantities and is defined as:

$$J = w_1 * RMS + w_2 * ME + w_3 * ST \tag{4.1}$$

with $w_1$, $w_2$ and $w_3$ as weighting factors. In the simulations, $w_1$, $w_2$ and $w_3$ are chosen as 100, 10 and 0.1 to equally weight each parameter of the performance index so that no parameter is favored in the determination of the best solution. The RMS is defined as:

$$RMS = \left( \frac{\sum_{k=1}^{N} e_y^2(k)}{N} \right)^{1/2} \tag{4.2}$$

where N is the number of iterations in the simulation. The ME is the largest absolute value of the output error, $e_y$, and ST is the time it takes the output to settle to 4% of its maximum value.

<u>The Grid Search Procedure:</u> The grid search procedure is optional. The goal of the procedure is to characterize the performance surface and is typically done when searching for global instead of local minima. The grid search is only done once, when the disturbance

M. Peek and P. J. Antsaklis, "Parameter Learning for Performance Adaptation in Large Space Structures" (Invited), P roc. o f t he 4 th I EEE I nternational S ymposium o n I ntelligent C ontro l , Albany, NY, Sept. 25-27, 1989.

dictionary is empty. The procedure uses the initial parameters of S, given by the control design, to define the range of the search intervals. The grid search routine steps through the intervals to obtain an approximate mapping of the performance surface. The data collected during the evaluation process includes each parameter of S, the performance index, and all components of the performance index (RMS, ME and ST). This data is stored in a collimated format and sorted. The three minimum performance indices and the associated controller parameters are then used by the optimization search procedure to determine an optimal set of controller parameters. The final three sets of the optimization search are then stored in the disturbance dictionary for future use. After this initialization, the disturbance dictionary is consulted for a starting point anytime a new disturbance is encountered.

The optimization search procedure: The optimization search procedure is implemented using a variation of the Hooke and Jeeves multidimensional search algorithm [6]. Formulated as an unconstrained, nonlinear optimization problem, the search for the optimal parameters of the adaptive controller is based on the performance index presented in (4.1).

In the optimization search procedure, as is typically the case, it is assumed that the performance surface is strictly quasiconvex. The local versus global minimum problem is not addressed at this level. It is also assumed that there is no magnitude constraint on the values of the parameters, although precautions are taken in the rules of the knowledge-based system to guard against expanding the search too fast, thus keeping the system in a stable parameter region. The knowledge-based system alternates between an exploratory search and a multidimensional pattern search of the performance surface until an optimal controller parameter set is obtained according to predetermined criteria. It is important to note, however, that the mathematical relation between the parameters of the adaptive control system and the performance index is unknown. Furthermore, the optimization method does not directly utilize the gradient, which is also unknown.

The knowledge-based system contains the rules for changing the parameters of the adaptive controller. During the optimization process, the knowledge-based system monitors the control system performance by keeping track of the parameter currently being varied, the current step size and direction, and the next parameter of the search process. The Hooke and Jeeves algorithm was chosen for the parameter search because it has shown faster convergence than a cyclic search [6].

The results obtained by the search methods are very dependent on the initial step size, the starting points, and the stopping criteria. If the initial step size is increased, a larger portion of the performance surface is explored. However, this may lead to unstable behavior of the closed-loop adaptive system. The effect of the stopping criteria is similar. When made too large, the search for the minimum performance index will be cut short. Too small a stopping criteria, however, does not yield a significant decrease in the performance index, wasting search energy. Starting points also affect the relative coverage of the performance surface. It is possible with a different starting point to find a better minimum on the performance surface since the global minimum problem is not being addressed this level. This is the reason for implementing the grid search procedure. While an exhaustive search of the entire performance surface is unwieldy, it is possible to get a general impression of the performance surface to narrow the search space.

It should be noted that the Hooke and Jeeves method has been used for similar purposes in [8,9] to auto-tune control parameters of a robotic arm. The method presented here, although similar to some aspects of the Chen's work, differs in the addition of the grid search procedure and the disturbance dictionary to assist in the selection of initial starting points for the optimization search procedure.

The disturbance dictionary: The disturbance dictionary is initially empty and is gradually built from training data consisting of pulses of various magnitudes and durations. In the learning system, it is assumed that the capability of identifying the disturbance by these measurable quantities is present. When it has been determined that an external disturbance is causing poor performance, the dictionary is used to determine a starting set of controller parameters for the optimization search procedure, based on the magnitude and duration of the disturbance. For a certain operating region (reference model, initial conditions and inputs) the optimization search procedure finds an optimal set of parameters, $S_{opt}$. Once the optimal set of parameters is found, information related to the disturbance and the optimal parameter set, $S_{opt}$, is added to the disturbance dictionary for future use.

## 5 The Transient Regulation Experiment

In this experiment, the reference model of the adaptive system is set to zero throughout the simulation, i.e. it has zero input and zero initial states. The plant takes the form;

$$x_p(k+1) = (\Phi_p + \Gamma_p G)x_p(k) + \Gamma_p u_p(k) \tag{5.1a}$$
$$y_p(k) = C_p x_p(k), \tag{5.1b}$$

where the state feedback gain matrix G is obtained originally by using the LQR routine in CTRL-C, and then changed to a minimum state configuration while maintaining stability and the shape of the open loop response. Further details of the plant model are found in [5, 11, 14].

The adaptive control action is provided by

$$u_p = k\tilde{\gamma} \tag{5.2}$$

where $\tilde{\gamma}^T = [e_y \; x^T_m \; u_m]$, $u_p \in R$, $u_m \in R$, $e_y \in R$, $x_m \in R^r$, $k^T \in R^{(r+2)}$, and r is the dimension of the modelled system. The gain matrix k has the form

$$k = k_P + k_I \tag{5.3}$$

where $k_I$ and $k_P$ the integral and proportional gains, respectively, are given by;

$$\dot{k}_I = -\sigma_1 k_I + L e_y \tilde{\gamma}^T T \tag{5.4}$$
$$\dot{k}_P = -\sigma_2 k_P + \bar{L} e_y \tilde{\gamma}^T \bar{T} \tag{5.5}$$

The controller parameters in the set $\{\sigma_1, \sigma_2, L, \bar{L}, T, \bar{T}\}$ are to be chosen in the design process to optimize system performance. However, there is no systematic, analytical method available for designers to make the best possible choice. Stability and convergence analysis could possibly provide a stability bound on these values, but due to the complexity of the transient response analysis of this nonlinear system, the analytical relation between performance and the actual parameter values is extremely difficult to determine. Here we shall determine the optimal performance via a non-analytical, systematic method.

Without loss of generality, T and $\bar{T}$ are fixed and set to 0.05. The parameters $\sigma_1$, $\sigma_2$, L, and $\bar{L}$ are then optimized. For the remaining discussion, S will denote the parameter values $\{\sigma_1, \sigma_2, L, \bar{L}\}$ and $S_{opt}$ will denote the values of the parameters that optimize the system performance. Using the initial parameters given in the design of the adaptive control system [14], $\sigma_1 = 0.5$, $\sigma_2 = 21.99$, $L = \bar{L} = 1.0*10^4$, and introducing a pulse disturbance with a magnitude of 2.0 and duration of two seconds, the closed-loop adaptive control system is enabled to track the zero reference model output. The plant input and output are presented in Figure 3a and 3b. The performance index of the original adaptive control system and the components of the performance index, are given in Table 1.

Next, the simulation of the closed-loop adaptive control system is performed using the optimization search procedure of the parameter learning system. The plant input and output, after the optimization search procedures have found $S_{opt}$, are shown in Figure 4a and 4b. The performance improves using the modified Hooke and Jeeves algorithm by 20% in 82 iterations. The performance index, and the components of the performance index are given in Table 1. The optimal parameter set $S_{opt.}$ is $\sigma_1 = 0.0063$, $\sigma_2 = 0.086$ and $L = \bar{L} = 1.0*10^4$.

The second part of the parameter learning system, the grid search procedure, is then added to characterize the performance surface. The results of the grid search are coupled with the modified Hooke and Jeeves optimization search procedure to obtain a new $S_{opt}$. The plant input and output of this system are shown is Figure 5a and 5b. The performance index, and the components of the performance index, are given in Table 1. The optimal parameter set $S_{opt}$ is $\sigma_1 = 0.093$, $\sigma_2 = 10.05$, L = 49,000 and $\bar{L}$ = 119,703 providing 30% improvement over the original system design and 11.6% improvement in 100 iterations over only the modified Hooke and Jeeves algorithm. The number of total iterations required is 8,812, which includes the grid search procedure. The grid search does add considerable computational time to find the optimal parameter set, and the number of iterations of the modified Hooke and Jeeves algorithm has increased as well. However, it does increase the confidence of the parameter set found as being near optimal. In addition, the grid search is run only once, if needed, when the disturbance dictionary does not contain knowledge about the performance of the system.

After training the parameter learning system with the disturbances listed in Table 2, different disturbances known to cause instability in the original adaptive control system are introduced. These disturbances are listed in Table 3. The resulting optimal parameter sets, $S_{opt}$, found by the antenna parameter learning system are also presented in Table 3, along with the respective performance indices. Figures 6through 9 show the plant input and output for each of these disturbances both before the assistance of the parameter learning system, using the original controller parameters, and after the parameter learning system is added to assist the adaptive controller using the information learned from training. Note that the system always settles within 4% of its maximum value after learning. Additional unstable disturbances may also be controlled as long as the the parameter learning system is trained to handle them.

The results show that, besides learning the values of optimal controller parameters that improve system performance, the parameter learning system is able to extend the region of operability through training. Training is required to be incremental, building upon previous knowledge. Since the relationship between the performance index and the controller parameter set in non-linear, the learning process needs to take steps small enough to determine the effects of different size disturbances on the plant.

## 6 Conclusions

This method is general and was also successfully applied for verification purposes to determine optimum gain in an LQR problem. However, specialized methods are obviously more efficient to solve specialized problems. General methods, like the one presented here, are recommended to be used in complicated problems when traditional methods fail. The method presented is also modular, both the functional evaluation and the optimization search procedures can be modified to match the particular problem. In addition, functional evaluation can be performed via computer simulation, physical experimentation or mathematical calculation. The generality of the method also allows it to be extended to different types of systems, such as multi-input multi-output (MIMO) systems, or to learn different system inputs, such as command inputs for second order reference models [16].

Future research directions include the expansion of the dictionary used in connection with the learning system to include different plant models and controllers so that it can be used as a scheduler in an autonomous control system [3]. In addition, an Associative Memory structure implemented via neural networks is proposed to map new environmental and plant conditions to the control parameter values. In this way, already acquired knowledge can perhaps be used to characterize the control and plant environment more effectively.

**Acknowledgment:** This work was supported in part by the Jet Propulsion Laboratory, Pasadena, CA, under contract No. 957856.

**References:**

[1] Anderson, C.W., "Learning to Control an Inverted Pendulum with Connectionist Networks", *Proc. of the 1988 American Control Conference*, pp. 2294-2298, Atlanta, GA, June 15-17, 1988.

[2] Antsaklis, P.J., "Panel Discussion on: Machine Learning in a Dynamic World", *Proc. of the 3rd IEEE Intern. Symposium on Intelligent Control*, Arlington, VA, August 24-26, 1988.

[3] Antsaklis, P.J., Passino, K.M., and Wang, S.J., "Towards Intelligent Autonomous Control Systems: Architecture and Fundamental Issues", *Journal of Intelligent and Robotic Systems*, Vol. 3, No. 4, 1989.

[4] Arbib, M.A., *Brains, Machines, and Mathematics, 2nd Edition*, Springer-Verlag, New York, 1987.

[5] Bayard, D.S., "Averaging Methods for Transient Analysis in Adaptive Control of Large Space Structures", Jet Propulsion Laboratory Engineering Memo, EM 347-88-228, Feb. 4, 1988.

[6] Bazaraa, M.S., and Shetty, C.M., *Nonlinear Programming Theory and Algorithms*, John Wiley & Sons, New York, 1979.

[7] Chen, K. and Lu, S. C.-Y., "A Machine Learning Approach to the Automatic Synthesis of Mechanistic Knowledge for Engineering Decision-Making", *Proc. of the 1988 Fourth Conference on Artificial Intelligence Application*, pp. 306-311, San Diego, CA, March 14-18, 1988.

[8] Chen, Y., "A Formulation of Automatic Parameter Tuning for Robot Arms", *Proc. of the 26th Conference on Decision and Control*, pp. 595-598, Los Angeles, CA, December 9-11, 1987.

[9] Chen, Y., "Automatic Parameter Tuning and Its Implementation for Robot Arms", *Proc. of the 1988 Conference on Robotics and Automation*, pp. 691-698, Philadelphia, PA, April 24-29, 1988.

[10] Eldred, D.B., "System Model for JPL/AFAL Experimental Structure", Private Communication, January, 1988.

[11] Gao, Z., Peek, M. and Antsaklis, P.J., "Learning For the Adaptive Control of Large Flexible Structures", *Proc. of the 3rd IEEE Intern. Symposium on Intelligent Control*, Arlington, VA, Aug 24-26, 1988.

[12] Goldberg, D.E., "Computer-Aided Gas Pipeline Operation Using Genetic Algorithms and Rule Learning", Ph.D. dissertation, (Civil Engrg.), University of Michigan, Ann Arbor, 1983.

[13] Handelman, D.A., and Stengel, R.F., "Rule-Based Mechanisms of Learning for Intelligent Adaptive Flight Control", *Proc. of the 1988 American Control Conference*, Atlanta, GA, June 15-17, 1988.

[14] Ih, C.-H.C., Bayard, D.S., Wang, S.J., and Eldred, D.B., "Adaptive Control Experiment with a Large Flexible Structure", *Proc. of the AIAA Guidance, Navigation, and Control Conference*, August 1988.

[15] Michalski, R.S., Carbonell, J.G., and Mitchell, T.M., *Machine Learning-An Artificial Intelligence Approach*, Tioga, Palo Alto, 1983.

[16] Peek, M.D. "Parameter Learning for Performance Adaptation in Large Space Structures", MS Thesis, Depart. of Electrical and Computer Engrg., University of Notre Dame, Notre Dame, IN, August,1989.

[17] Stepp, R., "Conjunctive Conceptual Clustering: A Method and Experimentation", Ph.D. Thesis, Department of Computer Science, University of Illinois, Urbana, 1984.

[18] Vivian, H.C., et. al, "Flexible Structural Control Laboratory Development and Technical Demonstration: Final Report", *JPL Publication 88-29*, October 1, 1987.

| | RMS | ME | ST | J |
|---|---|---|---|---|
| Original Adaptive System: | 0.195 | 0.087 | 19.1 | 4.7315 |
| Modified Hooke and Jeeves: | 0.017 | 0.075 | 12.8 | 3.7633 |
| Grid + Hooke and Jeeves: | 0.015 | 0.058 | 12.15 | 3.3240 |

TABLE 1: Performance Indices

| Disturbance | | $S_{opt}$ | | | | Performance |
|---|---|---|---|---|---|---|
| Amplitude | Duration | $\sigma_1$ | $\sigma_2$ | L | $\bar{L}$ | Index |
| 2.0 | 2 | 0.093 | 10.05 | 49000 | 119703.96 | 3.324 |
| 2.25 | 2 | 0.213 | 7.035 | 39200 | 131674.35 | 3.969 |
| 2.5 | 2 | 0.302 | 7.35 | 44046.1 | 145910.16 | 4.509 |
| 3.0 | 2 | 0.307 | 15.786 | 37325.68 | 183327.84 | 5.39 |
| 3.5 | 2 | 0.876 | 9.555 | 44046.1 | 175092.19 | 5.956 |
| 4.0 | 2 | 0.808 | 10.482 | 29114.03 | 203493.9 | 6.415 |
| 4.5 | 2 | 1.767 | 10.482 | 32025.43 | 203493.9 | 6.897 |
| 5.0 | 2 | 3.924 | 8.695 | 48450.71 | 140073.75 | 7.394 |
| 5.5 | 2 | 6.928 | 7.732 | 41567.48 | 138395.55 | 7.871 |
| 6.0 | 2 | 11.08 | 10.05 | 41567.48 | 138395.55 | 8.437 |
| 7.0 | 2 | 14.41 | 19.1 | 41567.48 | 110716.44 | 9.723 |

TABLE 2: Training Disturbances and Parameter Sets

M. Peek and P. J. Antsaklis, "Parameter Learning for Performance Adaptation in Large Space Structures" (Invited), Proc. of the 4th IEEE International Symposium on Intelligent Control, Albany, NY, Sept. 25-27, 1989.

| Disturbance | | $S_{opt}$ | | | | Performance |
|---|---|---|---|---|---|---|
| Amplitude | Duration | $\sigma_1$ | $\sigma_2$ | L | $\tilde{L}$ | Index |
| 8.0 | 2 | 23.06 | 19.1 | 41567.48 | 88573.15 | 10.81 |
| 9..0 | 2 | 1383.5 | 11.33 | 48450.71 | 77923.24 | 11.526 |

TABLE 3: Unstable Disturbances and Resulting Parameter Sets

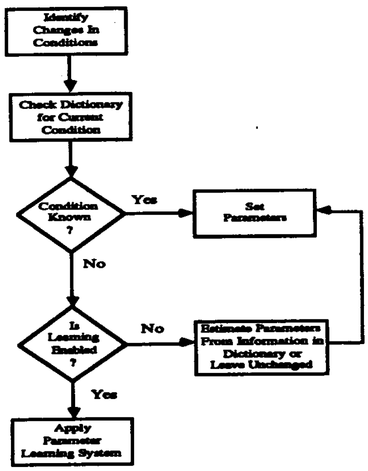



FIGURE 1: Functional Diagram for Parameter Setting

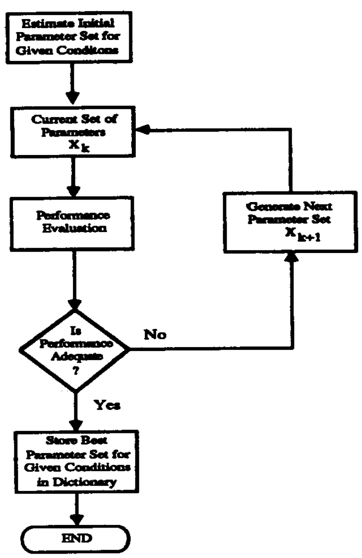



FIGURE 2: Functional Diagram of the Parameter LearningSystem

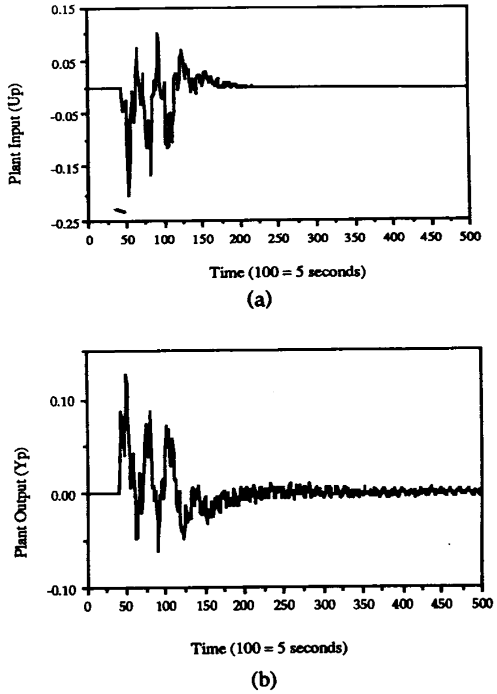



FIGURE 3: Plant Input and Output of Original Control System

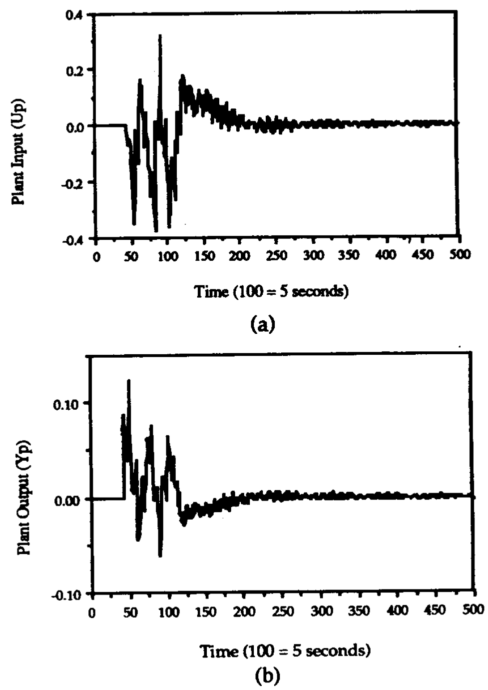
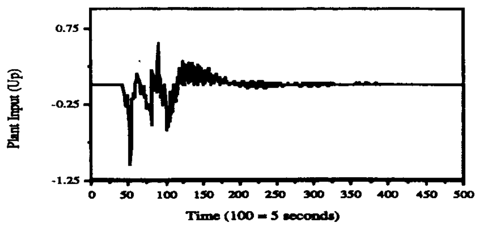
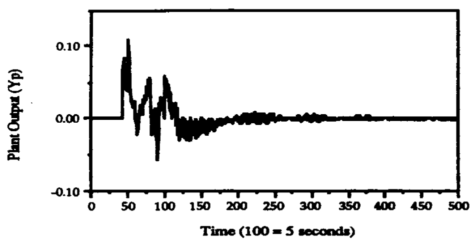



FIGURE 4: Plant Input and Output after Modified Hooke and Jeeves Algorithm

M. Peek and P. J. Antsaklis, "Parameter Learning for Performance Adaptation in Large Space Structures" (Invited), P roc. o f t he 4 th I EEE I nternational S ymposium o n I ntelligent C ontro l , Albany, NY, Sept. 25-27, 1989.

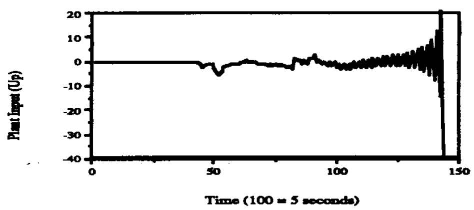



(a)

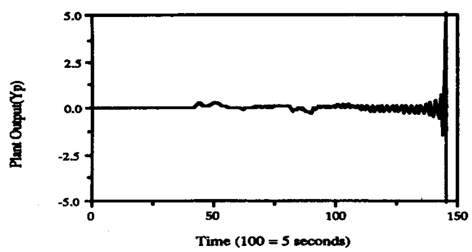



(b)

FIGURE 5: Plant Input and Output after Modified Hooke and Jeeves Algorithm coupled with Grid Search

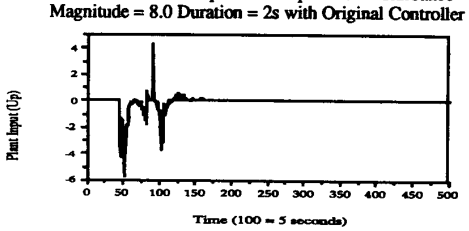



(a)

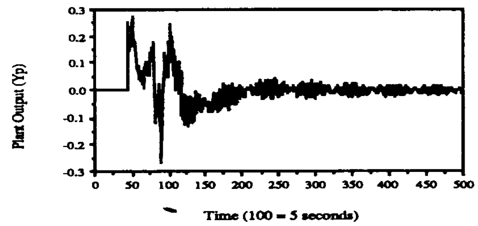



(b)

FIGURE 6: Plant Input and Output with Disturbance Magnitude = 8.0 Duration = 2s with Original Controller

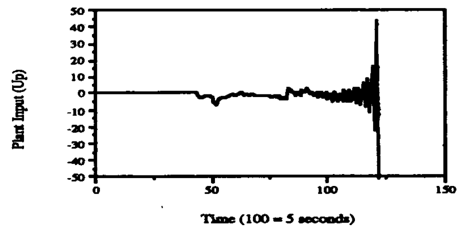



(a)

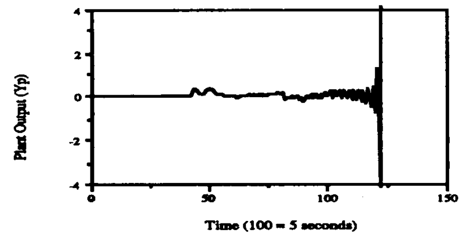



(b)

FIGURE 7: Plant Input and Output with Disturbance Magnitude = 8.0 Duration = 2s with Learned Parameters

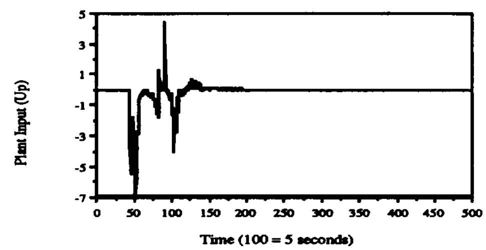



(a)

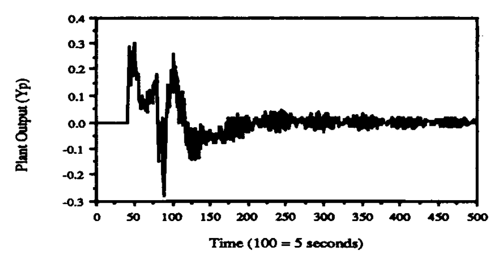



(b)

FIGURE 8: Plant Input and Output with Disturbance Magnitude = 9.0 Duration = 2s of Original Controller




(a)




(b)

FIGURE 9: Plant Input and Output with Disturbance Magnitude = 9.0 Duration = 2s with Learned Parameters