*Reprinted from:* PATTERN RECOGNITION AND MACHINE LEARNING
Edited by K. S. Fu
*(Plenum Press, 1971)*

# ADAPTIVE MODEL CONTROL APPLIED TO REAL-TIME BLOOD-PRESSURE REGULATION

Bernard Widrow

Stanford University

Stanford, California, U.S.A.

## ABSTRACT

A real-time computer-control system for regulating the blood pressure of an animal in a prolonged state of shock has been successfully developed and is being theoretically analyzed. The computer controls the rate of infusion of a vaso-constrictor drug inputed to the animal, and monitors the blood-pressure output. An adaptive model of the animal's drug response is used to derive the required input for control of future blood-pressure values. A transversal-filter model is used, and control is derived by forward-time calculation including the known internal states of the model.

There is a great need for learning control systems which can adapt their control laws to accomodate the requirements of plants whose characteristics may be unknown and/or changeable in unknown ways. A principal factor that has hampered the development of adaptive controls is the intrinsic difficulty of dealing with learning processes embedded in feedback loops. Interaction beteeen the feedback of the learning processes and that of the signal flow paths greatly complicates the analysis which is requisite to the design of dependable operating systems.

An elementary form of adaptive control system employing an adaptive series compensator is shown in Figure 1. This system is simple in conception but is rather inefficient and difficult to deal with from the point of view of adaptation. The compensator could be easily adapted if one had available in real time an optimal output or plant driving signal corresponding to the particular real-time compensator input signal. The optimal compensator output

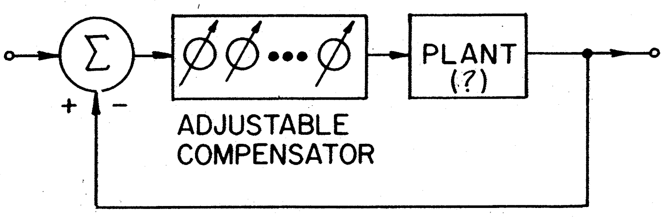


Figure 1. Adaptive Series-Compensated Control System.

signal could serve as a training signal for an adaptive compensator. This signal is very difficult to obtain when the plant is unknown however. If this signal were available, the compensator and the feedback loop would be unnecessary.

Another approach to the adaptation of the system of Figure 1 is the following. Suppose that the purpose of adaptation is the minimization of the servo error in the mean square sense. Gradient components could be measured by perturbing the compensator adjustments. The mean square error could be minimized by using a gradient method such as the method of steepest descent. There are two difficulties here that limit the technique. Regardless of the method used in perturbing the adjustments, whether one at a time or all at once, the system settling time must be waited before measurements can be taken each time the compensator adjustments change or the plant parameters change.[1]. Furthermore, assuming that the gradient can be successfully measured, the mean-square-error performance function is known to be irregular, non-parabolic, and containing relative optima [2]. Hill-climbing techniques for such functions still are in primitive stages of development.

The techniques proposed in this paper have been successfully tested in a limited number of medical-electronic experiments and they represent a different approach to plant control which circumvents many of the difficulties typified by the adaptive-system example of Figure 1. These techniques are still in development, so this paper should be regarded as a preliminary report. In some ways, these techniques are related to those of Powell [3] who used an adaptive model to determine a feedback-loop compensator.

The techniques proposed here will be referred to as Adaptive-Model Control (AMC). The principle operates as follows. Form a model of the plant, and continually update the model by an adaptive process. Using the model and its internal states, do a forward-time analysis to determine inputs to the model which will cause desired future model outputs, thereby controlling the model very closely. Apply the same control to the actual plant, and if the model behaves similarly to the plant, the output of the plant will be closely controlled. The control of the plant is in a sense open-loop, but in fact, the loop is closed through the adaptive process.

To illustrate the adaptive-Model Control, an overall diagram of a blood pressure control system that has been constructed and tested is presented in Figure 2.

At the beginning of a test, a quantity of the powerful drug Arfonod is injected into the animal (a dog). This drug has the effect of disabling the natural blood pressure regulating system of the animal inducing a prolonged state of shock. If left alone, the blood pressure would drop close to zero and irreversible damage would be done to the animal. A vasoconstrictor drug, Norepinephrine, is infused slowly over many hours to compensate and to support the blood pressure. The computer continually monitors blood pressure and regulates the rate of infusion of the vasoconstrictor. The ultimate purpose is to develop computer controls for human intensive care systems.

Typical dynamic responses of the mean animal blood pressure readings to step changes in rate of infusion of the vasoconstrictor drug are sketched in Figure 3. The type of response resulting depends upon the size, type, and especially upon the condition of

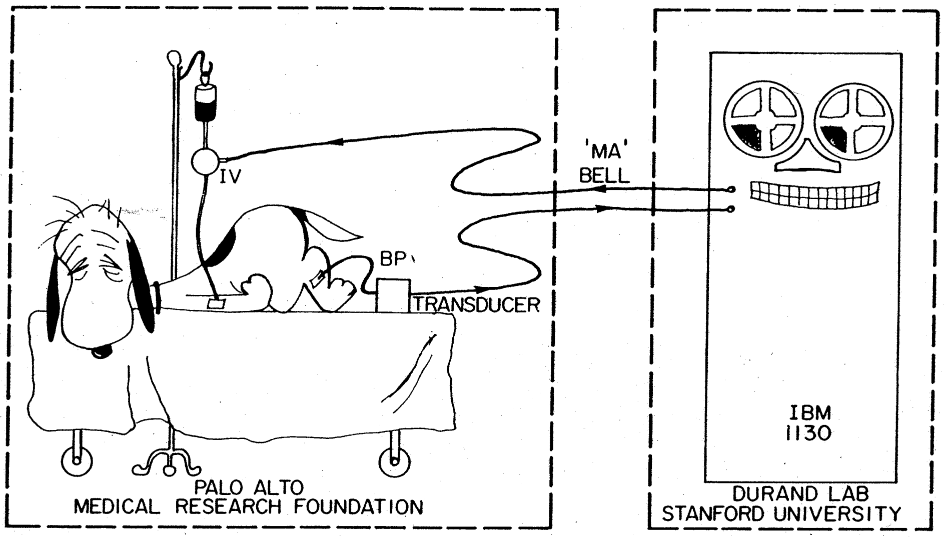


Figure 2. The Experimental Set-up.

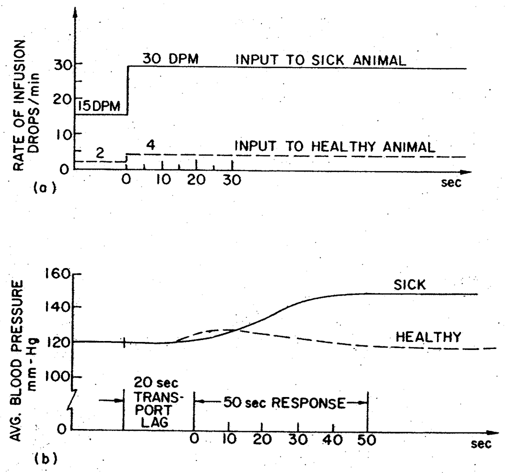


Figure 3. Typical Average Blood Pressure Time Responses to Step Changes in Vasoconstrictor Infusion Rates.

the animal. An animal in good health will respond to small increases in drug flow by eventually settling the blood pressure back to the original set point level. A sick animal will not be able to compensate for even moderate increases in vasoconstrictor inputs and hence the blood pressure will increase in a predictable manner and then settle at a higher level. Tremendous variations in animal responses to the vasoconstrictor have been observed. Typically, there is a transport lag of 10 to 20 seconds before the animals respond, and settling times are usually about 50 to 100 seconds.

The system illustrated in Figure 2 gives the appearance of being an ordinary feedback control system. But this is not the case. The dynamic response of an animal (including transport lag) is too variable to be managed by a conventional feedback control. A block diagram of the actual system is shown in Figure 4.

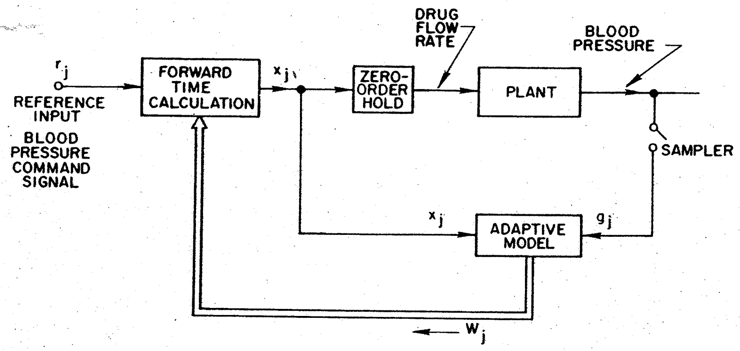


Figure 4. Block-Diagram of an Adaptive-Model Control System.

The functions labeled "forward-time calculation" and "adaptive model" are accomplished by a Hewlett-Packard 2116 computer, as are many data logging and data display function which are not shown but which are necessary in a laboratory set-up. The "plant" is the dynamic response of the blood pressure system to the drug. The zero-order hold is part of the electronic system interfacing the computer to the drug-flow solenoid valve. A cycle time occurs every five seconds. Once per cycle, the adaptive model is updated and a new drug rate (drops per minute) is calculated.

The adaptive model is a 20-tap transversal filter covering a total real-time window of 95 seconds. A bias weight is included to represent the ambient average blood pressure when the input drug rate is zero. The details of the adaptive model are shown in Figure 5.

The adaptive model would be linear if the weights were fixed or if their values were not functions of the input-signal characteristics. The adaptive process automatically adjusts the weights so that for the given input-signal statistics, the model provides a best minimum-mean-square-error fit to a sampled version of the combination of the zero order hold and the plant. The adaptive process utilized is the LMS algorithm, presented first in refs. [4] and [5] and presented more completely in the context of applications to pattern recognition [6] and applications to spatial and temporal filtering (adaptive antenna arrays) [7].

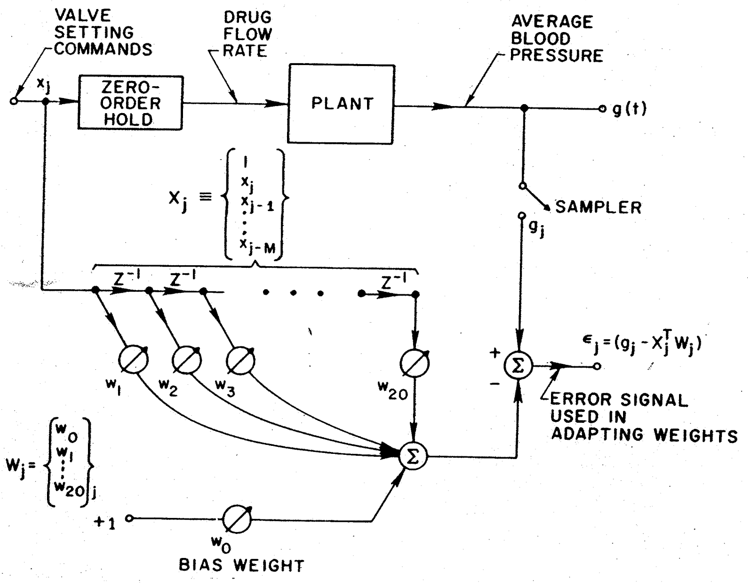

VALVE SETTING COMMANDS
DRUG FLOW RATE
AVERAGE BLOOD PRESSURE
$x_j$
ZERO-ORDER HOLD
PLANT
g(t)
SAMPLER
$g_j$
$X_j \equiv \begin{Bmatrix} 1 \\ x_j \\ x_{j-1} \\ \vdots \\ x_{j-M} \end{Bmatrix}$
$z^{-1}$
$z^{-1}$
$z^{-1}$
$w_1$
$w_2$
$w_3$
$w_{20}$
$\Sigma$
+
−
$\Sigma$
$\epsilon_j = (g_j - X_j^T W_j)$
ERROR SIGNAL USED IN ADAPTING WEIGHTS
$W_j = \begin{Bmatrix} w_0 \\ w_1 \\ \vdots \\ w_{20} \end{Bmatrix}_j$
+1
$w_0$
BIAS WEIGHT

Figure 5. Details of "Adaptive Model" Box, a 20-tap Adaptive Transversal Filter with Bias Weight.

The adaptive algorithm is

$$W_{j+1} = W_j + 2\mu\varepsilon_j X_j \ , \qquad \varepsilon_j = d_j - X_j^T W_j \ . \tag{1}$$

The error $\varepsilon_j$ is the difference between the desired response $d_j$ and the model response $X_j^T W_j$. The desired response $d_j$ is obtained by sampling the actual plant response g(t). Therefore $d_j = g_j$, and the error is $\varepsilon_j = g_j - X_j^T W_j$. (Refer to Figure 5.)

With a stationary input and a stationary plant, the LMS algorithm is known to be convergent when the convergence factor $\mu$ is chosen in the range

$$\frac{1}{\lambda_{max}} > \frac{1}{\text{trace } R} > \mu > 0 \ . \tag{2}$$

The input autocorrelation matrix R is defined below. Its largest eigenvalue is $\lambda_{max}$. Note that trace R = $E[X_j]^2$. The factor $\mu$ controls stability and rate of convergence of the algorithm. The expected value of the weight vector converges to the optimal or "Wiener" solution $W^*$.

$$\lim_{j\to\infty} E[W_j] = W^* = R^{-1}P,$$

where

$$R \equiv E[X_j X_j^T] \quad \text{and} \quad P \equiv E[d_j X_j] \ . \tag{3}$$

A fundamental mathematical question is raised by this approach. The input cannot be stationary, and it will be shown that this input is partly determined from the weight values themselves (via the "forward-time calculation"). Yet, the LMS algorithm behaves stably, and in all cases in practice, always converges rapidly to a close model of the unknown plant. There seem to be no practical problems with the approach, only mathematical problems.

Refer now to the block diagram of the entire system shown in Figure 4. The plant control $X_j$ is derived from the box labeled "forward-time calculation." This box generates $X_j$ from the reference input $r_j$ and from the weight vector $W_j$ and the input vector $X_j$ of the model. We shall now consider the operation of this box.

The objective is to derive a driving function $x_j$ so that $X_j^T W_j = r_j$. Each iteration cycle, the model weight vector $W_j$ is updated, and then $x_j$ is calculated taking into account $W_j$ and

$x_{j-1}, x_{j-2}, \ldots, x_{j-n+1}$. If the model behavior is essentially the same as that of the plant, application of $x_j$ to the plant will cause its output response to be close to the reference command signal $r_j$. Let us choose $x_j$ according to [1].

$$x_j w_1 + \sum_{i=2}^{n} x_{j-i+1} w_i + w_0 = r_j$$

$$x_j = \frac{1}{w_1} [r_j - w_0 - \sum_{i=2}^{n} x_{j-i+1} w_i] \qquad (4)$$

Choosing $x_j$ according to this formula will allow the model to be perfectly controlled, and applying the same input to the actual plant will result in a mean-square control error $E[r_j - g_j]^2$ equal to the mean-square error $E[\varepsilon_j^2]$ of the modelling process.

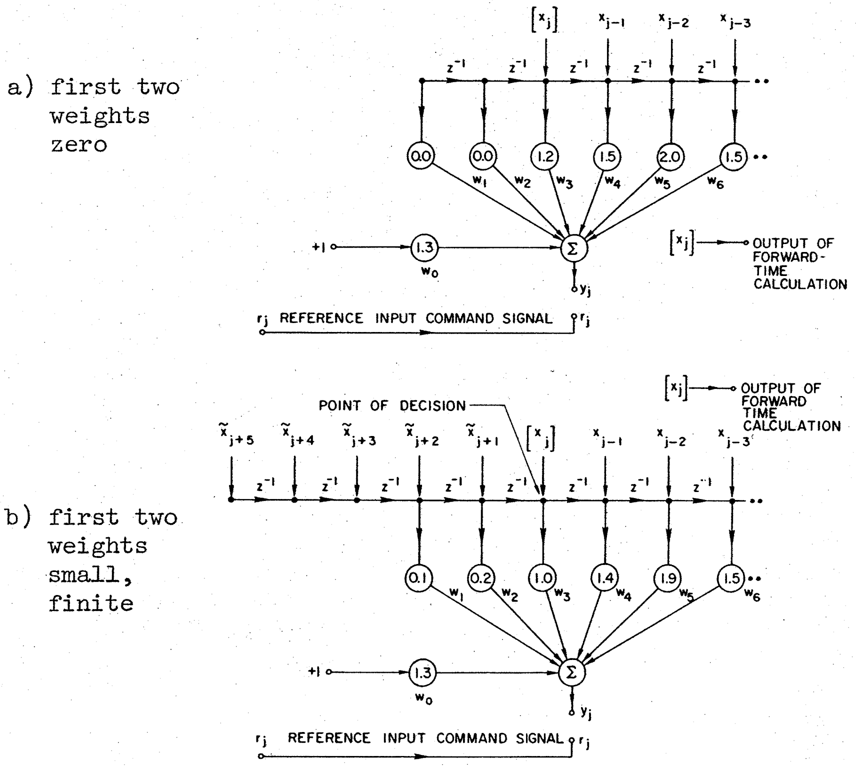
Everything goes well using this method as long as $w_1$ has substantial value. When there are transport delays however, $w_1$ tends to be small and noisy. The values of $x_j$ computed by the above formula could be very large and erratic, since division by $w_1$ is required. This could create problems, particularly in the blood-pressure control system where massive doses of drug are undesirable and negative doses are impossible. Because of transport delays, two somewhat different approaches have been taken.

The first of these approaches constrains the first several weights of the adaptive model to be zero. The number of zero-constrained weights corresponds to the transport lag of the plant, which would be obtained from a priori knowledge.

Details of the functional box "Forward-time Calculation" of Figure 4 are shown in Figure 6a, illustrating how $x_j$ is calculated in the situation when the first two model weights are zero. All the weights shown in Figure 6 are copied from the values derived by the appropriate modeling process. The particular values shown are for illustration only.

Each cycle, the value of $x_j$ is calculated to cause the output of the summer $y_j$ to be equal to $r_j$. Since the plant is driven by $x_j$, its sampled output $g_j$ will closely approximate $r_{j-2}$, depending on the closeness of fit of the model to the plant. The delay in the response is an inevitable result of the plant transport delay. The values of $x_j$ are calculated according to

$$x_j = \frac{1}{w_3} [r_j - w_0 - \sum_{i=4}^{n} x_{j-i+3} w_i] . \qquad (5)$$



a) first two weights zero

b) first two weights small, finite

Figure 6. Details of "Forward-Time Calculation" Box.

It should be mentioned that to start one of these systems as quickly as possible, initial weight values in the modeling process are usually taken from the previous run. Initial values are not critical, but if they are close to correct, there will be very little start-up transient.

The second approach for dealing with plant transport delay does not require a decision constraining a certain number of model weights to zero. There are many cases where the leading weights are small in magnitude, but non-zero. Such a set of weights is illustrated in Figure 6b.

In this case, the values of $x_j$ cannot be calculated to perfectly match the output $y_j$ with $r_j$. Future tentative values of x such as $\tilde{x}_{j+1}$, $\tilde{x}_{j+2}$... are calculated so that the control signal $x_j$ can be deduced. The tilde indicates that the values are tentative. Absence of the tilde means that the value is decided and is used in controlling the plant. The "point of decision" in the calculation is indicated in Figure 6b. The position of this point of decision along the tapped delay line of the adaptive model is chosen a priori by the system designer to correspond to the plant transport delay. Choosing this position has some effect on system performance, but is not critical.

The calculation of $x_j$ at the time of the jth cycle is accomplished according to eq. (6). The point of decision is taken as in Fig. 6b.

$$w_0 + w_1\tilde{x}_{j+2} + w_2\tilde{x}_{j+1} + w_3x_j + \sum_{i=4}^{n} w_ix_{j-i+3} = r_j$$

$$w_0 + w_1\tilde{x}_{j+3} + w_2\tilde{x}_{j+2} + w_3\tilde{x}_{j+1} + w_4x_j + \sum_{i=5}^{n} w_ix_{j-i+4} = r_{j+1}$$

$$w_0 + w_1\tilde{x}_{j+1} + w_2\tilde{x}_{w+3} + w_3\tilde{x}_{j+2} + w_4\tilde{x}_{j+1} + w_5x_j + \sum_{i=6}^{n} w_ix_{j-i+5} = r_{r+2}$$

The number of equations is generally determined by the number of future values of the reference command signal $r_j, r_{j+1}, r_{j+2}, \ldots$ that may be available. These equations may be rearranged according to (7).

$$w_3x_j + w_2\tilde{x}_{j+1} + w_1\tilde{x}_{j+2} = r_j - w_0 - \sum_{i=4}^{n} w_ix_{j-i+3}$$

$$w_4x_j + w_3\tilde{x}_{j+1} + w_2\tilde{x}_{j+2} + w_1\tilde{x}_{j+3} = r_{j+1} - w_0 - \sum_{i=5}^{n} w_ix_{j-i+4}$$

$$w_5x_j + w_4\tilde{x}_{j+1} + w_3\tilde{x}_{j+2} + w_2\tilde{x}_{j+3} + w_1\tilde{x}_{j+4} = r_{j+2} - w_0 - \sum_{i=6}^{n} w_ix_{j-i+5} \quad (7)$$

The numerical values of the right-hand sides of the equations (7) can be calculated since $r_j$, $r_{j+1}, r_{j+2}, \ldots$ are known, the weights are known, and the previously-decided driving function values $x_{j-1}, x_{j-2}, \ldots$ are known. Let the right-hand sides be calculated.

These equations cannot be solved yet, since there are too many "unknowns" for the number of equations. Since $w_1$ and $w_2$ are relatively small, a solution can be obtained by letting two adjacent distant-future values of $\tilde{x}$ take arbitrary values, such as zero. When only three values of the reference signal $r_j, r_{j+1}, r_{j+2}$ are known, we have three simultaneous equations to solve. We let $\tilde{x}_{j+4} = \tilde{x}_{j+3} = 0$. It is then possible to solve for $x_j, \tilde{x}_{j+1}$, and $\tilde{x}_{j+2}$.

Although we only need the decided value $x_j$ at the time of the jth cycle for direct control purposes, the future tentative values a interesting to have also.

At the j+1-th cycle, the entire process is repeated. The tentative value of $\tilde{x}_{j+1}$ calculated on the j-th cycle should agree closely with the decided value of $x_{j+1}$ calculated on the j+1-th cycle. The agreement will not be perfect because of setting $\tilde{x}_{j+4}$ and $\tilde{x}_{j+3}$ to zero. Let us call this effect "truncation error." By using additional future values of the reference command signal, tentative values of x's can be determined further into the future and truncation error can be reduced.

Using $x_j$ as the plant driving function will cause the plant sampled output $g_j$ to agree closely with $r_{j-2}$. The error in the system response will be due partly to imperfection in the modeling fit and partly to truncation error.

Note that when the transport delay mechanism is such that the first k weights of the model are relatively small, solving the equations determining $x_j$ and future $\tilde{x}$-values requires assuming that a sequence of k distant-future $\tilde{x}$-values are zero.

Also note that knowledge of future values of the plant driving function, although they are tentative, could be quite useful in modifying the goals of the control system (i.e., modifying $r_j$) in cases where demands are made on the driving function that would exceed limits, go negative where this is not possible, etc. For example, the sequence $r_j$ could be modified by not insisting that the system settle in the minimum time achievable with an unrestricted $r_j$, etc. It is possible to state and to have the system respond to very sophisticated computer-directed goals. Since inexpensive modern computers can operate much faster than real time, various goals and control objectives can be practically explored each calculation cycle.

The AMC techniques have already been used a number of times in experiments to regulate and control average blood pressure in animals. In these experiments, the standard deviation of the noise in the blood-pressure sensing instrumentation has been about 5 to 10 mm Hg. The mean blood pressure is typically regulated to within about 2 to 4 mm Hg in steady state and could be off about 5 to 10 mm Hg temporarily under extreme transient conditions. The typical start-up settling times are of the order to two minutes, somewhat longer than the total time window spanned by the adaptive plant model. The Appendix presents data from an actual run.

## APPENDIX - AN EXPERIMENTAL RUN

Figures 7-10 present results developed during an experimental run while controlling an animal's average blood pressure.

The beginning of the run is shown in Figure 7. The dog was healthy and normal until the Arfonod was injected, whereupon his blood pressure plummetted, as seen in the figure.

In this experiment, the adaptive-model weights began to form, starting from initial settings, at the very outset before the Arfonod was injected. The two upper tracings show the actual, average blood pressure of the animal and the output of the model respectively. Note how they stay closely together. They stay moderately close together even in periods of great stress such as just after the Arfonod was injected.

At the beginning of the run, the flow rate of the vasoconstrictor (the "drug rate") was manually set at 10 drops per minute. This was manually raised to 20 drops per minute after the Arfonod took hold. Raising the drug rate checked the blood pressure decline. Soon thereafter, as indicated by the cross on the drug-rate tracing, the control of drug rate was turned over to the automatic system and remained automatic thereafter. A pressure set point was entered through the computer keyboard, and this level was indicated by the cross near the upper two tracings. The control system then had the job of getting the animal blood pressure up to the set point and holding it there in spite of natural disturbances in the animal. Changes in the set point were inserted from time to time as part of the system test. The middle curve shows a running average mean square error (on a log scale) between the plant and the adaptive model.

The total memory time of the adaptive model was 100 seconds. The model contained 20 taps with 5 second delays between taps of the transversal filter. Once automatic control was established, the system took about 5 minutes to settle the blood pressure close to the set point. Thus the system settling time was about 3 times as long as the memory time of the model. This represents rather fast settling for an adaptive control system.

In this test, the computer controlled the blood pressure during several hours with the animal under different degrees of influence to Arfonod. The results were uniformly good, and the response data of Figures 8 and 9 typical. The data records were long, but the data of Figures 7, 8 and 9 are contiguous with slight time overlaps. Settling responses to changing set-point values are illustrated. In each case, approximately 5 minute settling times were evident.

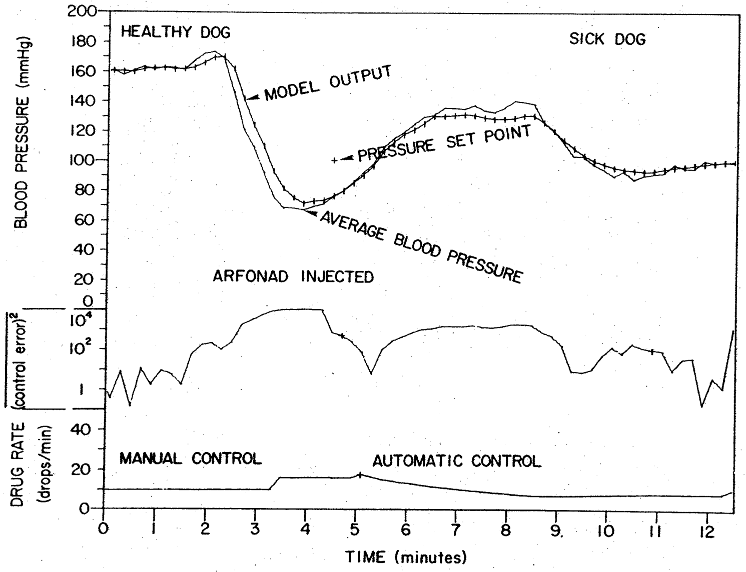


Figure 7. Actual run: Transition from Healthy to Sick, Manual to Automatic Control.

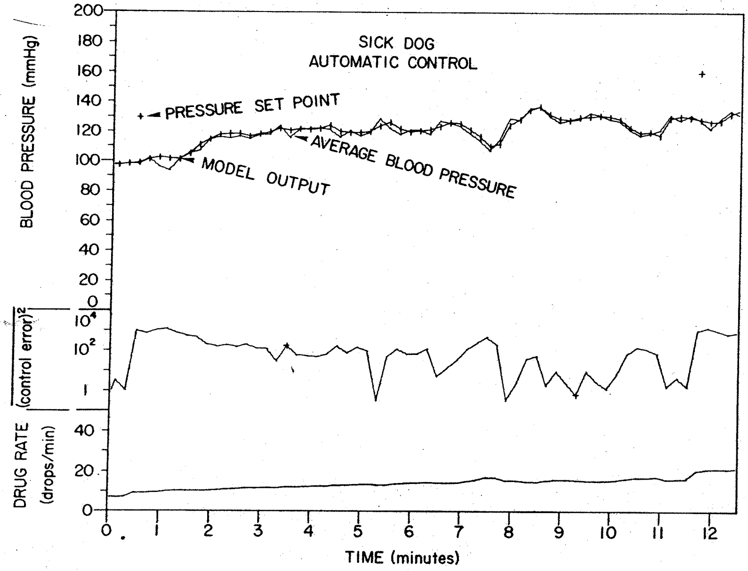


Figure 8. Actual run: Control of Sick Dog Blood Pressure.

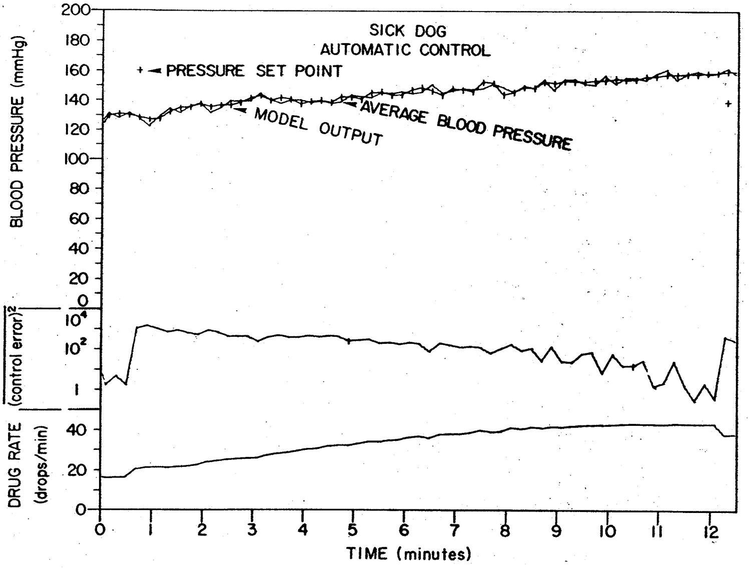


Figure 9. Actual run: Control with Raised Blood-Pressure Set Point.

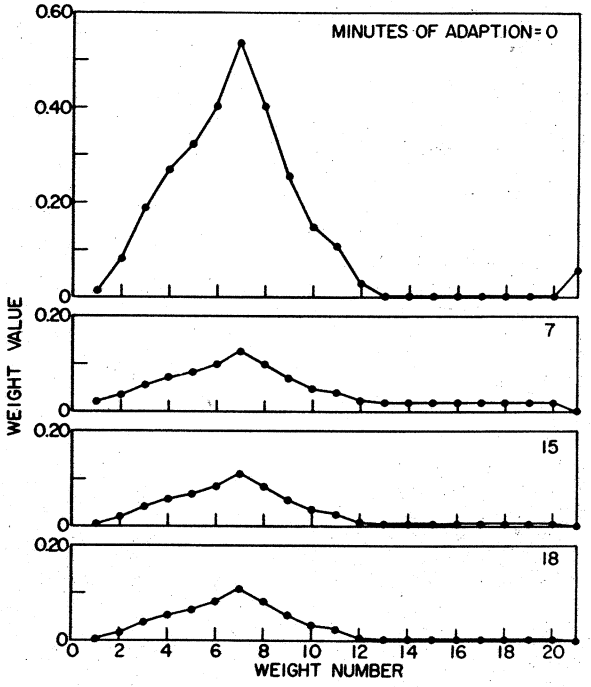


Figure 10. Model Weights (Impulse Response).

The tapped-delay-line-model weights at several times during the run were recorded and they are plotted in Figure 10. The weight values are arranged chronologically along the line, and so represent the model's view of the animal's impulse response. The twenty-first weight is the bias weight, (see Figure 5). The impulse response of the top frame was taken before the Arfonod was injected and here the animal was very sensitive to the vasoconstrictor drug. The next frame was taken after the Arfonod was injected and had taken hold, just before the automatic control was turned on. The shape of the response was changed somewhat, but the sensitivity level changed greatly. As time went on, the animal regulating system was disabled due to the Arfonod. Changes in the animal impulse response took place, but they were not drastic changes. One can see that the amount of transport delay was not very clear-cut, illustrating the type of plant behavior indicated in Figure 6b.

Although this system is simple in conception, making the hardware and software work reliably has taken considerable effort. Many runs over the past year or so were made in order to perfect the algorithms, the system software, and the interface hardware. Experimental results have been uniformly good using the algorithms and procedures outlined here. The development of mathematical analysis of the AMC adaptive control technique is progressing nicely.

## REFERENCES

1. K. S. Narendra, and L. E. McBride, "Multiparameter Self-Optimizing Systems Using Correlation Techniques", IEEE Transactions on Automatic Control, Vol. AC-9, No. 1, January 1964, p. 31.

2. P. E. Mantey, "Convergent Automatic-Synthesis Procedures for Sampled-Data Networks with Feedback", Stanford Electronics Laboratories, SEL-64-112, TR 6773-1, Stanford, Calif., Oct. 1964.

3. F. D. Powell, "Predictive Adaptive Control", IEEE Transactions on Automatic Control, Vol. AC-14, No. 5, October 1969, pp. 550-552.

4. B. Widrow and M. E. Hoff, "Adaptive Switching Circuits," 1960 IRE Wescon Convention Record, Part 4, pp. 96-104.

5. B. Widrow, "Adaptive Sampled-Data Systems," Proceedings, 1st International Congress IFAC, Moscow, 1960, pp. 406-411.

6. J. Koford and G. Groner, "The Use of an Adaptive Threshold Element to Design a Linear Optimal Pattern Classifier," IEEE Transactions on Information Theory, Vol. II-12, No. 1, January 1966, pp. 42-50.

7. B. Widrow, et. al., "Adaptive Antenna Systems," Proceedings, IEEE, Vol. 55, No. 12, December 1967, pp. 2143-2159.

## ACKNOWLEDGEMENTS

The author would like to thank Henry Crichlow, Fred Mansfield, Jeffrey Bower, Thomas Stibolt, Ulf Strom, and especially Cristy Schade who was the leader of this group of graduate students at Stanford University who have designed, assembled, and brought to operation the real-time experimental hardware and software. Special thanks go to Dr. Noel M. Thompson of the Palo Alto Medical Research Foundation who introduced the problem, made working with experimental animals possible, and who was able to evaluate performance from the point of view of both a Physician and an Engineer.