

Fig. 1. Block diagram of the DA SPSA neural network-based control system.

## A Diagonal Recurrent Neural Network-Based Hybrid Direct Adaptive SPSA Control System

Xiao D. Ji and Babajide O. Familoni

**Abstract**—A direct adaptive simultaneous perturbation stochastic approximation (DA SPSA) control system with a diagonal recurrent neural network (DRNN) controller is proposed. The DA SPSA control system with DRNN has simpler architecture and parameter vector size that is smaller than a feedforward neural network (FNN) controller. The simulation results show that it has a faster convergence rate than FNN controller. It results in a steady-state error and is sensitive to SPSA coefficients and termination condition. For trajectory control purpose, a hybrid control system scheme with a conventional PID controller is proposed.

**Index Terms**—Diagonal recurrent neural network (DRNN), neural network controller (NNC), simultaneous perturbation stochastic approximation (SPSA).

### I. INTRODUCTION

Nonlinear adaptive control system design is a challenge in nonlinear control system theory. In general, one may use neural networks (NN) to identify and/or control unknown and/or uncertain nonlinear

Manuscript received May 9, 1996; revised March 6, 1998. Recommended by Associate Editor, J. C. Spall.

The authors are with the Department of Electrical Engineering, The University of Memphis, Memphis, TN 38152 USA (e-mail: familoni@memphis.edu).  
Publisher Item Identifier S 0018-9286(99)04545-6.

systems. To accomplish this, one needs to train (usually, offline) an inverse neural network as a controller. This is generally difficult since the system is unknown. An ideal scheme is a direct adaptive (DA) neural network control system. Spall described a generalized NN based on the simultaneous perturbation stochastic approximation (SPSA) approach to estimate the gradient of the performance function of an unknown nonlinear system [1]. Such a direct adaptive SPSA approach does not require any prior knowledge of the unknown system and does not need a separate training phase. An SPSA direct adaptive control system will converge to an optimal neural network parameter set, if it exists [3]. The NN-based SPSA approach as discussed in [2] and [3] uses a forward neural network (FNN) as the controller. The parameter vector size, in general, is large. For example, a network which contains four layers, two inputs, one output, with two hidden layers containing 20 and 10 nodes, respectively (denoted as  $N_{2,20,10,1}^4$ ), has 280 elements in the parameter vector. Other things being equal, its increased computational cost results in a slow performance measurement period (i.e., sampling period), and the performance measurement period is very important for a real-time control application.

As is well known, a recurrent neural network (RNN) has some advantages over FNN such as faster convergence, more accurate mapping ability, etc., but it is difficult to apply the gradient-descent method to update the neural network weights in RNN [4]. Ku *et al.* [5], [6] proposed the DRNN scheme that captures the dynamic behavior of a system and, since it is not fully connected, training is expected to be much faster than RNN. DRNN with time delay has RNN behavior but simple connections and is easy to use when applying the gradient-descent method. Therefore, in this paper, a diagonal recurrent neural network (DRNN) is employed in a DA SPSA control system. Simulation results are compared with those of the FNN SPSA scheme. These results also show that in general, after the SPSA process, the fixed DA SPSA neural network-based control results in a steady-state error because of the finite sample constraint of the SPSA approach. To improve the control performance, a conventional PID controller was employed to form a hybrid DA SPSA scheme. The proposed hybrid DA SPSA control system was examined by simulation and showed good performance.

### II. SPSA BACKGROUND

Consider the problem of finding a root  $\theta^*$  of the gradient equation

$$g(\theta) \equiv \frac{\partial L(\theta)}{\partial \theta} = 0 \quad (1)$$

for some differentiable cost function  $L : R^p \rightarrow R^1$ . There are many methods for finding  $\theta^*$ . In the case where  $L$  is observed in the presence of noise, an SA algorithm of the generic

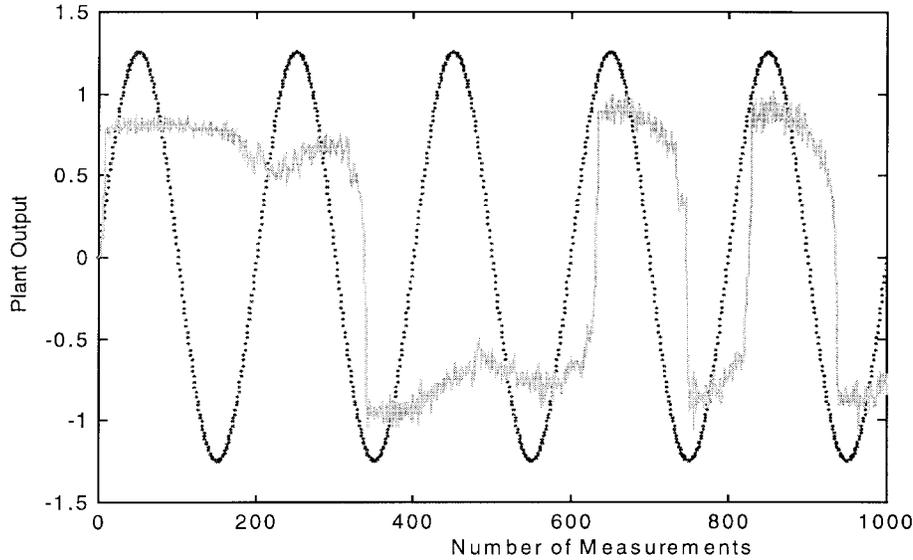


Fig. 2. System output as a function of output measurements using SPSA with a  $N_{2,20,10,1}^4$  FNN as NNC. The solid line denotes the actual system output  $y_p$ , while the dotted line denotes the desired system output,  $y_d$ .

Kiefer–Wolfowitz/Blum-type is appropriate [7], [8]. It is based on finite difference methods that require  $2p$  (noisy) measurements of  $L$  at each iteration. The estimated  $\hat{\theta}$  at the  $(k+1)$ th iteration is

$$\hat{\theta}_{k+1} = \hat{\theta}_k - a_k \hat{g}_k(\hat{\theta}_k) \quad (2)$$

where  $\hat{g}_k(\cdot)$  is the estimated gradient at the  $k$ th iteration, and the gain sequences are calculated by  $a_k = a(k+A)^{-\alpha}$ , where  $a$ ,  $A$ , and  $\alpha$  are nonnegative coefficients [9].

In the SPSA method,  $\hat{g}_k(\cdot)$  is estimated by the “simultaneous perturbation” method as follows. Let  $\Delta_k \in R^p$  be a vector of  $p$  mutually independent mean-zero random variables  $\{\Delta_{k1}, \Delta_{k2}, \dots, \Delta_{kp}\}$  satisfying certain important conditions [1], [7]. Furthermore, let  $\{\Delta_k\}$  be a mutually independent sequence with  $\Delta_k$  independent of  $\hat{\theta}_0, \hat{\theta}_1, \dots, \hat{\theta}_k$ . We make two measurements

$$y_k^{(+)} = L(\hat{\theta}_k + c_k \Delta_k) + \varepsilon_k^{(+)} \quad (3a)$$

$$y_k^{(-)} = L(\hat{\theta}_k - c_k \Delta_k) + \varepsilon_k^{(-)} \quad (3b)$$

where  $\varepsilon_k^{(+)}$  and  $\varepsilon_k^{(-)}$  represent measurement noise terms that satisfy

$$E(\varepsilon_k^{(+)} - \varepsilon_k^{(-)} \mid f_k, \Delta_k) = 0 \quad \text{a.s. } \forall k$$

and where  $f_k \equiv \{\hat{\theta}_0, \hat{\theta}_1, \dots, \hat{\theta}_k\}$ . With a gain sequence  $c_k = c(k+1)^{-\gamma}$ , where  $c$  and  $\gamma$  are nonnegative coefficients, the estimation of  $g(\cdot)$  at the  $k$ th iteration is

$$\hat{g}(\hat{\theta}_k) = \begin{bmatrix} \frac{y_k^{(+)} - y_k^{(-)}}{2c_k \Delta_{k1}} \\ \vdots \\ \frac{y_k^{(+)} - y_k^{(-)}}{2c_k \Delta_{kp}} \end{bmatrix}. \quad (4)$$

The members of  $\{\Delta_k\}$  are chosen here with a Bernoulli  $\pm 1$  distribution with a probability of  $1/2$  for each outcome. The name “simultaneous perturbation” as applied to this method arises from the fact that all elements of the  $\hat{\theta}_k$  vector are being varied simultaneously. If there is little change in several successive iterations, the algorithm

is terminated. The last iterate  $\hat{\theta}_k$  is designated an estimate of the optimum  $\theta^*$ . Note that this estimate only needs two measurements of  $L$  instead of  $2p$  in the usual finite difference approximation.

Reference [1] showed that under certain conditions the bias in  $\hat{g}_k(\cdot)$  as an estimate of  $g(\cdot)$  goes to zero as  $k \rightarrow \infty$ , and  $\hat{\theta}_k$  converges almost surely (a.s.) to  $\theta^*$ . Reference [9] has already described the basic steps for implementing SPSA. Pragmatic and theoretically valid values for  $\alpha$  and  $\gamma$  are 0.602 and 0.101, respectively. The value of  $c$  can be set at a level approximately equal to the standard deviation of the measurement noise in  $y(\theta)$ . The value for  $a$  is typically found by experimentation, usually starting at a value of one and decreasing or increasing this value if, respectively, the algorithm seems to be behaving erratically or too conservatively [9]. Adjustments to  $a$  and  $c$  can be made by monitoring the first few iterations on the problem of interest or by testing the algorithm on a similar (perhaps reduced dimension) problem.

### III. DIRECT ADAPTIVE SPSA CONTROL SYSTEM WITH DRNN

References [2] and [3] proposed two kinds of adaptive control systems with SPSA algorithm: direct adaptive control and self-tuning adaptive control. When virtually nothing is known about the plant, the direct approximate or adaptive (DA) control approach is appropriate but the self-tuning adaptive control (STA) requires that some prior information exist about the plant. Fig. 1 shows block diagrams of the DA scheme employed in our study.

An unknown nonlinear plant,  $y(k+1) = 0.8 \times \sin(2y(k)) + 1.2 \times u(k)$ , previously employed by other authors, was used as the test plant in this study [4], [10]. The desired plant output is a sinusoidal wave. FNN and DRNN neural network controllers were employed for comparison purposes. In both cases, the  $\tanh(\cdot)$  function was employed as a sigmoidal function and each iteration involved two measurements (samplings).

In the FNN case, the size of the parameter vector  $\hat{\theta}$  is  $p = 280$  with biases in hidden layers only. The SPSA algorithm with FNN needs 250 multiplication operations, 280 addition operations, and 30 sigmoidal function calculations for each  $y$  measurement. Simulation results show that the FNN with SPSA takes longer to simulate each measurement and the plant output converges slower than the

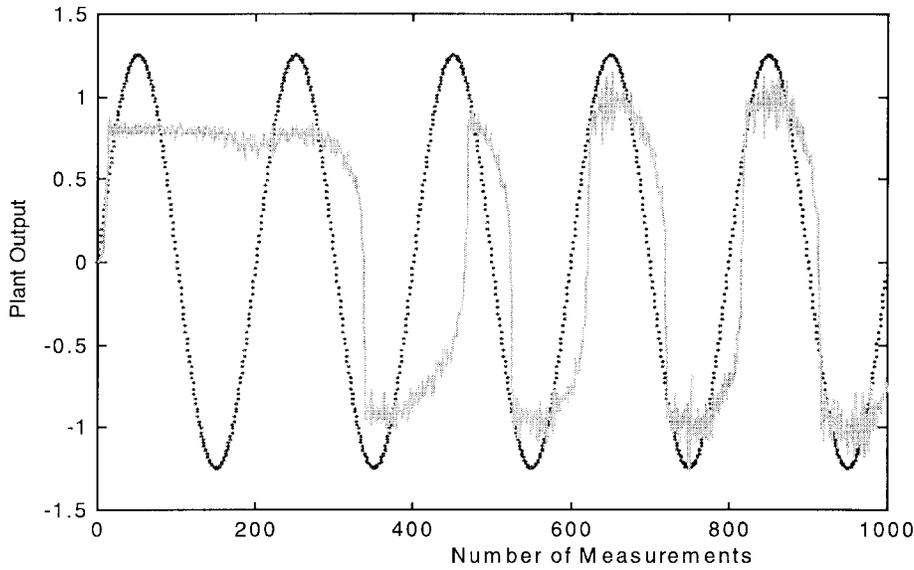


Fig. 3. System output as a function of output measurements using SPSA with a  $\mathcal{N}_{2,5,1}^3$  DRNN as NNC. The solid line denotes the actual system output  $y_p$ , while the dotted line denotes the desired system output,  $y_d$ .

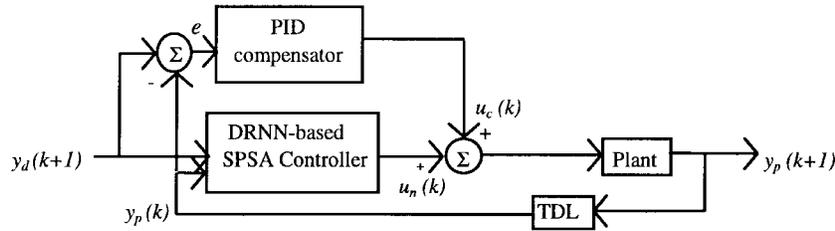


Fig. 4. A hybrid control scheme: DA SPSA neural network-based control system with PID compensator.

DRNN with SPSA does. In Fig. 2, the first 1000 measurements of a simulation with the SPSA coefficients  $\alpha = 0.602$ ,  $\gamma = 0.101$ ,  $a = 0.775$ , and  $c = 0.1$  are shown. The termination condition was  $\|\Delta\theta\| < 0.01$  for 100 successive measurements.

In the DRNN case, using the same notation as the FNN case, the NNC is  $\mathcal{N}_{2,5,1}^3$ , and the size of the parameter vector  $\hat{\theta}$  is  $p = 25$  with biases. The SPSA algorithm with DRNN needs 20 multiplication operations, 20 addition operations, and 5 sigmoidal function calculations for each measurement. Different configurations of DRNN ( $\mathcal{N}_{2,7,1}^3, \mathcal{N}_{2,9,1}^3$ ) were investigated and the results show no significant difference. Therefore, we choose the simplest one as the appropriate one in our study. We performed a simulation of the DRNN case with the same termination condition as in the FNN case. After 1500th measurements, the system satisfied the requisite condition and the SPSA was terminated. Since FNN and DRNN have different net architectures with different parameters, it is difficult to compare their convergence speeds directly, but the computational cost for each measurement shows that the DRNN converges faster than the FNN. Fig. 3 shows the final results of the DRNN system after the SPSA. As pointed out in [11], there are steady-state errors in both cases, with the DRNN system having a larger steady error than the FNN system.

#### IV. A HYBRID DIRECT ADAPTIVE SPSA CONTROL SYSTEM

As mentioned above, after the SPSA algorithm the direct adaptive SPSA control system may have a steady-state error due to finite-sample considerations and limitations of the NN structure. Thus, at termination, the SPSA neural network-based controller has converged nearly to the optimal equilibrium point in the parameter space, and

the system output is close to the desired system output. Therefore, at this point a linear controller was employed to compensate for the error. The hybrid control scheme shown in Fig. 4 was proposed. For the trajectory control case, a conventional PID controller may be connected to the fixed SPSA neural network-based controller in parallel after the SPSA algorithm. The control force then becomes  $u = u_n + u_c$  where  $u_n$  is the fixed SPSA neural network controller output and  $u_c$  is the PID compensator output. The conventional PID algorithm is  $u_c(k) = b_0 \times e(k) + b_1 \times e(k-1) + b_2 \times e(k-2) + u_c(k-1)$ , where  $e(k)$  is the error signal and  $b_0$ ,  $b_1$ , and  $b_3$  are designed constants. In the presented case, we implemented the proportional component of the PID by setting  $b_0 = 0.19$ , and  $b_1 = b_2 = 0$ .

Figs. 5 and 6 show the simulation results of this proposed hybrid control scheme. The SPSA neural network-based controllers in both cases have the same SPSA algorithm coefficients ( $\alpha = 0.602$ ,  $\gamma = 0.101$ ,  $a = 0.775$ , and  $c = 0.1$ ) but with different termination conditions. Fig. 5 shows the simulation result of the proposed hybrid control system with termination condition  $\|\Delta\theta\| < 0.0005$  for 100 successive measurements. The system satisfied the termination condition after 8062 measurements with zero initial parameter vector. Fig. 6 shows the simulation result of the proposed hybrid control system with termination condition  $\|\Delta\theta\| < 0.001$  for 50 successive measurements. The termination point number is the 1534 measurements with zero initial parameter vector. These simulation results show that with the proposed hybrid control scheme the termination condition of the neural network-based SPSA DA control system may be relaxed while reaching almost the same control performance.

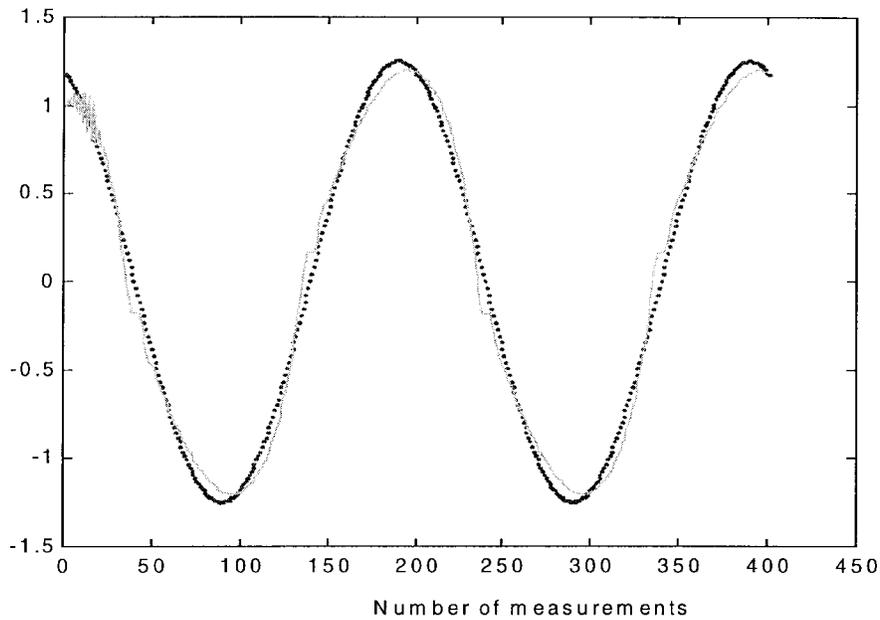


Fig. 5. System output of the hybrid DA SPSA control system after SPSA operation. SPSA operation met the termination condition ( $\|\Delta\theta\| < 0.0005$  for 100 successive measurements) at 8062 measurements. The solid line denotes the actual system output  $y_p$ , while the dotted line denotes the desired system output,  $y_d$  for measurements 8062 to 8512.

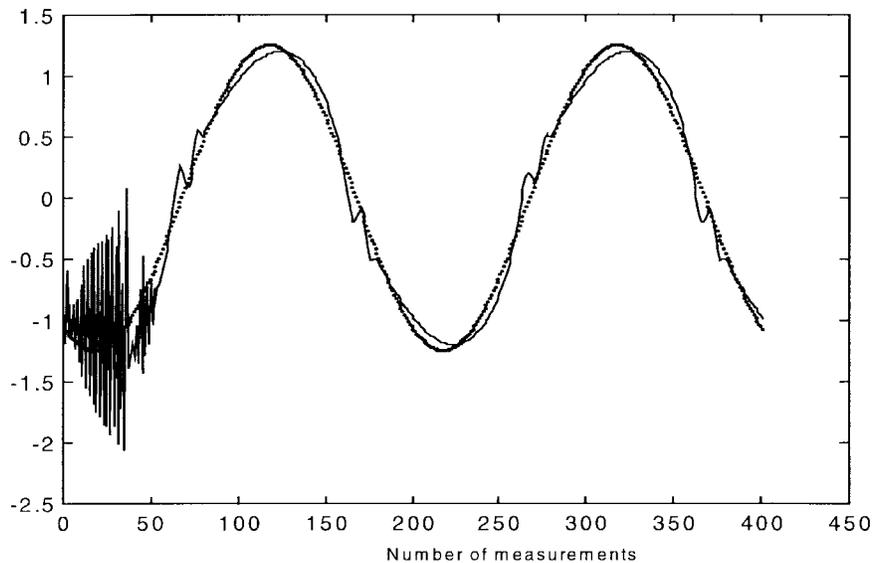


Fig. 6. System output of the hybrid DA SPSA control system after SPSA operation. SPSA operation met the termination condition ( $\|\Delta\theta\| < 0.001$  for 50 successive measurements) at 1534 measurements. The solid line denotes the actual system output  $y_p$ , while the dotted line denotes the desired system output,  $y_d$  for measurements 1534 to 1984.

## V. CONCLUSION

This short study confirms that the SPSA approach is appropriate for direct adaptive control scheme in the sense of statistical modeling and control. It can be applied to an unknown nonlinear system without the need to construct either a forward or an inverse model of the system. The DA SPSA system is stable and converges to an optimal state under certain conditions. The numerical experiments in this study suggest that use of DRNN for SPSA controller has the advantages of a simpler network architecture, reduced size of the parameter vector, and faster convergence rate than FNN controller.

Our study showed that for the trajectory control problem, SPSA has a steady-state error because of the finite iteration number. A good performance after SPSA also depends on the termination condition. The

risk of reaching a local minimum is inherent in most realistic adaptive estimation schemes and is not unique to SPSA. Although such local minima may be good enough for certain practical applications, the initial parameter vector must be close to the optimal parameter vector if it is necessary to avoid being stuck in a local minimum. The unknown global optimal parameter vector may be found using the SPSA alone, but system performance may possibly be improved by combining it with other methods (such as genetic algorithm, AI control, knowledge-based, etc.) as upper level control components.

Although only a proportional controller was implemented in this study, we suggest that the performance of this hybrid DA SPSA control scheme may be improved with a conventional PID controller as compensator. It should improve the control performance and relax the SPSA termination condition [12].

## REFERENCES

- [1] J. C. Spall, "Multivariate stochastic approximation using a simultaneous perturbation gradient approximation," *IEEE Trans. Automat. Contr.*, vol. 37, pp. 332-341, Mar. 1992.
- [2] J. C. Spall and J. A. Cristion, "Nonlinear adaptive control using neural networks: Estimation with a smoothed form of simultaneous perturbation gradient approximation," *Statistica Sinica*, vol. 4, pp. 1-27, 1994.
- [3] ———, "Model-free control of nonlinear stochastic systems with discrete-time measurements," *IEEE Trans. Automat. Contr.*, vol. 43, pp. 1198-1210, Sept. 1998.
- [4] A. Karakasoglu, S. I. Sudharsanan, and M. K. Sundareshan, "Identification and decentralized adaptive control using dynamic neural networks with application to robotic manipulators," *IEEE Trans. Neural Networks*, vol. 4, pp. 919-930, Nov. 1999.
- [5] C.-C. Ku and K. Y. Lee, "Diagonal recurrent neural network-based control: Convergence and stability," in *Proc. American Control Conf.*, Baltimore, MD, June 1994, pp. 3340-3345.
- [6] ———, "Diagonal recurrent neural networks for dynamic systems control," *IEEE Trans. Neural Networks*, vol. 6, pp. 144-155, Jan. 1995.
- [7] J. R. Blum, "Multidimensional stochastic approximation methods," *Ann. Math Stat.*, vol. 25, pp. 737-744, 1954.
- [8] J. Kiefer and J. Wolfowitz, "Stochastic estimation of a regression function," *Ann. Math Stat.*, vol. 23, pp. 462-466, 1952.
- [9] J. C. Spall, "Implementation of the simultaneous perturbation algorithm for stochastic optimization," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 34, pp. 817-823, 1998.
- [10] F. C. Chen, "Back-propagation neural networks for nonlinear self-tuning adaptive control," *IEEE Contr. Syst. Mag.*, pp. 44-48, Apr. 1990.
- [11] X. D. Ji and B. O. Fiamloni, "Experimental study of direct adaptive SPSA control system with diagonal recurrent neural network controller," in *Proc. IEEE Southeastcon '96*, Tampa, FL, pp. 525-528.
- [12] D. C. Chin, "A more efficient global optimization algorithm based on Stablinski and Tang," *Neural Networks*, vol. 7, no. 3, pp. 573-574, 1994.