High-Speed, Model-Free Adaptive Control Using Parallel Synchronous Detection

Dimitrios N. Loizos
Electrical and Computer Engineering
The Johns Hopkins University
Baltimore, MD 21218
dloizos@jhu.edu

Paul P. Sotiriadis
Electrical and Computer Engineering
The Johns Hopkins University
Baltimore, MD 21218
pps@jhu.edu

Gert Cauwenberghs
Division of Biological Sciences
University of California, San Diego
San Diego, CA 92093
gert@ucsd.edu

ABSTRACT

A VLSI implementation of an adaptive controller performing gradient descent optimization of external performance metrics using parallel synchronous detection is presented. Real-time model-free gradient estimation is done by perturbation of the metrics' control parameters with narrow-band deterministic dithers resulting in fast adaptation and robust performance. A fully translinear design has been employed for the architecture, making the controller operation scalable within a very wide range of frequencies and control bandwidths, and, therefore customizable for a variety of systems and applications. Experimental results from a SiGe BiCMOS implementation are provided demonstrating the broadband and high-speed performance of the controller.

Categories and Subject Descriptors

B.7.1 [**Integrated Circuits**]: Types and Design Styles—*algorithms implemented in hardware, VLSI (very large scale integration)*

General Terms

Algorithms, design, measurement, performance

Keywords

model-free, adaptive control, synchronous detection, multi-dithering, VLSI optimization, high-speed control, translinear circuits

1. INTRODUCTION

Several applications in RF and optical communications call for compensation of high speed variations, mainly in the propagating media, such as multi-path fading, co-channel interference [1] and atmospheric turbulence [2]. The nonlinear dynamics of the communication system, as well as the stochastic nature of the noise introduced by the propagating medium, preclude the use of conventional system identification and optimal control techniques, while speed requirements dictate real-time designs for the control. Adaptive schemes must be employed and dedicated VLSI solutions have to be considered.

Model-free architectures [3] can further enhance the robustness of the controller as well as the speed of adaptation. Instead of an exact model for the adaptive controller, which is impossible to have for a VLSI implementation due to mismatches in the fabrication procedure, knowledge of only a simple performance metric of the system can be used, over which optimization will be performed. The metric is optimized over several controllable parameters $u_i,\ i = 1, \ldots, n$ by perturbing each of them with a small amplitude dither. The perturbed portion of the metric is proportional to its gradient and this information is retrieved and used by the controller to update each of the parameters $u_i,\ i = 1, \ldots, n$ according to the gradient descent flow algorithm.

Several VLSI architectures/implementations using stochastic dithers have been proposed in the literature [4]-[7]. Although such solutions are computationally very efficient, they inherently suffer from limited optimization speeds, since the gradient is probabilistically measured *on the average*. In this work, a different approach has been followed, using deterministic dithers, leading to *true* gradient descent flow optimization and therefore higher adaptation speeds. The parameters u_i are perturbed with small amplitude sinusoidal signals of frequencies $\omega_i,\ i = 1, \ldots, n$, different for each parameter

$$\tilde{u}_i = u_i + \alpha \cos(\omega_i t).$$

Synchronous detection is performed parallely, between the perturbed performance metric and the corresponding dither for each parameter, in order to retrieve the gradient information[1]

$$\overline{J(\tilde{\mathbf{u}}) \cos(\omega_i t)} = \frac{\alpha}{2} \left. \frac{\partial J}{\partial u_i} \right|_{\mathbf{u}}.$$

Finally, all parameters are parallely updated according to the following rule,

$$\frac{du_i}{dt} = -G \mathrm{sgn} \left(\left. \frac{\partial J(\mathbf{u})}{\partial u_i} \right|_{\mathbf{u}} \right), \tag{1}$$

where G is the gain of the update. A difference between the above rule and the actual gradient descent flow algorithm is the use of the signum of the gradient instead of its exact

[1]Overline denotes low-pass filtering.

SBCCI '07, September 3-6, 2007, Rio de Janeiro, RJ, Brazil

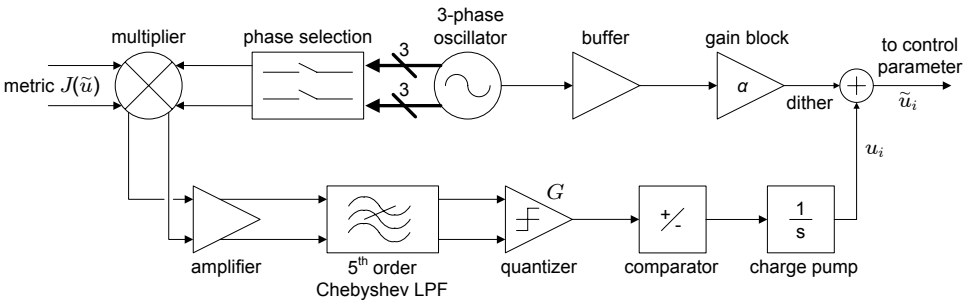


Figure 1: System architecture of a single channel.

value. Apart from the fact that such a rule is easier to implement in VLSI, this choice is also made to achieve faster convergence near the optimum state as well as higher immunity to time delays in the system, as will be shown in the next section.

2. SYSTEM ARCHITECTURE

For each parameter, a dedicated channel is used where the dither is added and synchronous detection is performed [8]. A block diagram of the channel architecture is shown in Fig. 1. The dither is generated by a 3-phase sinusoidal oscillator and one of its phases (fixed) is added to the control parameter u_i. All perturbed control parameters $\tilde{u}_i,\ i = 1, \ldots, n$ are applied to the under optimization metric J and its output is provided back to the input of all channels. After synchronous detection and since only the sign information of the gradient is needed for the gradient descent flow algorithm (eq. (1)), the output of the low-pass filter is quantized before driving a charge pump that updates the control parameter value u_i.

The necessity of a multi-phase oscillator becomes clear when delays are considered in the closed-loop [9]. Due to the narrow-band nature of the dithers, time delays τ_i are mapped to phase delays φ_i, which, through synchronous detection, cause the gradient information for each channel to be scaled by a factor $\cos(\varphi_i)$

$$\overline{J(\tilde{\mathbf{u}})_{\tau-\text{delay}}\cos(\omega_i t)} = \frac{\alpha}{2}\left.\frac{\partial J}{\partial u_i}\right|_{\mathbf{u}}\cos(\varphi_i),\ \ \varphi_i = \omega_i \tau_i.$$

Choosing an appropriate phase of the oscillator for synchronous detection, the error in gradient estimation due to time delays in the loop can be minimized. Since only the signum information of the gradient is used to update the values of the controlled parameters, only phase delays of $\varphi_i \in \left[\frac{\pi}{2} + 2\kappa\pi, \frac{3\pi}{2} + 2\kappa\pi\right]$ will cause a wrong update.

3. STABILITY ANALYSIS

The controller is modelled according to Fig. 2. The "sgn" block represents quantization and the integrator models the charge pump. Without loss of generality, the metric is considered as a generalized square function, form that represents typical cost functions, such as power. In order to simplify the stability analysis, the time/phase delays in the loop are considered known and compensated for by appropriate selection of the phase in the 3-phase oscillator.

Given any smooth cost function $J(\mathbf{u})$ with a global minimum or maximum and assuming update of its parameters according to rule (1), the Lyapunov stability theorem [10] states asymptotic convergence of J with time to its minimum/maximum, since

$$\frac{d}{dt}J(\mathbf{u}(t)) = -G\sum_{i=1}^{n}\left|\frac{\partial J}{\partial u_i}\right| < 0$$

for any $\mathbf{u}$'s other than the equilibrium points of J. However, in (1), possible phase shifts to the gradient information that can be introduced by the low-pass filter have not been taken into account. Such phase shifts may cause limit cycles, i.e. oscillations of the control variables $u_i, i = 1, \ldots, n$ around the equilibrium points. In order to investigate this possibility, the harmonic balance concept is applied; limit cycles at a frequency ω_H are assumed and by deriving the response of the system it is checked whether they are indeed sustained or not and if yes, what is the exact frequency ω_H. Each block of the system (Fig. 2) is replaced by its describing function, i.e., its response for a narrow band excitation at frequency ω_H.

The main results of the stability analysis are

- The controller's state converges to a limit cycle.
- The limit cycle occurs at a frequency at which the phase introduced by the filter is $\phi_H = -\frac{\pi}{2}$.

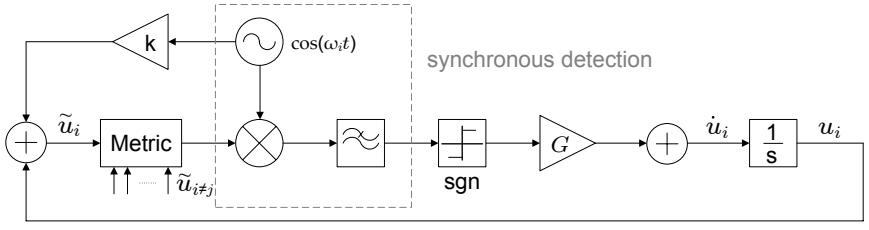


Figure 2: Model of a single channel for stability analysis.

- The amplitude of the limit cycle depends on the cut-off frequency of the filter and the gain of the charge pump.
- There exists a trade-off between convergence rate and amplitude of the limit cycle.

Although limit cycles occur, they are controllable and their amplitude can be kept low.

4. CIRCUIT DESIGN AND IMPLEMENTATION

The main challenge in the circuit implementation of the architecture is to design it in such a way that it can be used in a variety of applications, each with different bandwidth specifications. The design has to be, therefore, tunable in a wide range of frequencies and extremely broadband. To this end, translinear implementations are chosen for the tunable parts of the circuit and a 0.5μm SiGe BiCMOS process for its fabrication. An extra challenge is imposed by the lack of high-speed pnp devices in the process, leading to an all-npn translinear design.

4.1 Oscillator

The oscillator is designed as a ring of 3 differential $G_m - C$ filters in shunt with tunable resistors R, as shown in Fig. 4(a). The transfer function of each $G_m - R - C$ block has an amplitude of $|H(j\omega)| = G_m R/\sqrt{1+(\omega RC)^2}$ and phase given by $\tan^{-1}(H(j\omega)) = -\omega RC$. According to the Barkhausen oscillation criteria, the conditions for oscillation will be

$$R = \frac{\alpha}{G_m},\ \alpha = 2 \tag{2}$$

$$\omega = \frac{\sqrt{3}G_m}{2C} \tag{3}$$

Transconductance G_m is implemented as a simple differential pair (transistors Q_7 and Q_8). In order to satisfy (2), which mandates resistance R to scale inversely proportionally to G_m for oscillations to be sustained, R is taken as the emitter-base resistance of transistors Q_9 and Q_{10}, equally sized and biased as Q_7 and Q_8. The scaling factor $\alpha = 2$ in (2) is achieved by adding a gain stage of 2 (Q_1-Q_6) to prescale the input of the transconductance. In order to compensate for possible non-idealities that would reduce the gain below 2, the bases of Q_9 and Q_{10} are attached to the load of the gain stage, scaling α slightly above 2.

To add control to the actual value of α, current sources I_{AMP} are introduced. Denoting the tail current of both the gain and the transconductance stages as I_{FREQ}, the value of the scaling factor α is given by

$$\alpha = 3\frac{I_{FREQ}}{I_{FREQ} + 2I_{AMP}}$$

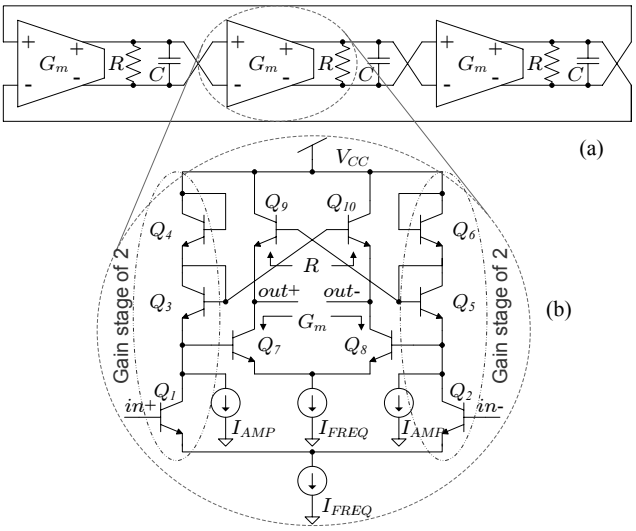


Figure 4: (a) High-level architecture of the 3-phase oscillator. (b) Circuit implementation of the $G_m - R$ blocks.

Increase in α leads to higher oscillation amplitudes but also higher THD (Total Harmonic Distortion) and vice versa.

Transconductance G_m is directly proportional to current I_{FREQ} and according to (3), the frequency of oscillation will also scale proportionally to I_{FREQ}. As will be shown later, current I_{FREQ} has been also used to bias the multiplier, while current I_{LPF} which biases the low-pass filter, is also directly related to I_{FREQ}. In this way, biasing, speed and power consumption scale uniformly for all tunable blocks, making possible a wide tuning range for the entire architecture.

4.2 Low-Pass Filter

Purpose of the low-pass filter is to block higher order intermodulation products from the output of the multiplier while keeping only the dc portion which is proportional to the gradient. The cut-off frequency of the filter sets the loop bandwidth of the system and, consequently, the maximum achievable adaptation speed. The trade-off between adaptation speed and attenuation of unwanted intermodulation terms is unavoidable.

A 5^{th} order Chebyshev filter with 1dB ripple is chosen for the design, since it provides steep roll-off and fairly constant gain at the pass band. The filter is implemented using biquads and 1^{st} order $G_m - C$ filters (Fig. 3). The cut-off frequency of the filter is tunable and set by the value of G_m. All transconductors G_m have the same topology and their gains are linearly controlled by replicas of the same current

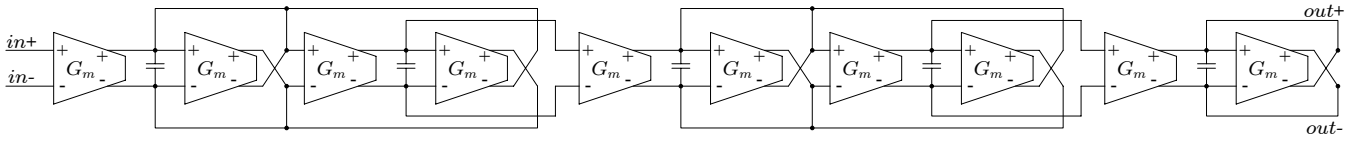


Figure 3: 5^{th} order Chebyshev low-pass filter using a tunable $G_m - C$ architecture.

I_{LPF}. The capacitors in the design are scaled according to the 5^{th} order Chebyshev polynomial.

4.3 Quantizer and Charge Pump

Purpose of the quantizer is to retrieve the sign information of the partial derivative of J with respect to u_i, i.e., $\frac{\partial J}{\partial u_i}$. Two were the main specifications for its design. First, it had to be fast enough to follow down to μs or faster changes in the gradient. Second, its offset should be absolutely smaller than the product of the dither with the perturbed portion of the gradient - typically a few hundredths of a Volt. The quantizer is implemented according to the design proposed in [11]. As can be seen in Fig. 5, the comparator has 3 stages: a low-offset pre-amplification, a high-speed latched decision circuit, and a differential to single-ended fully symmetrical output buffer.

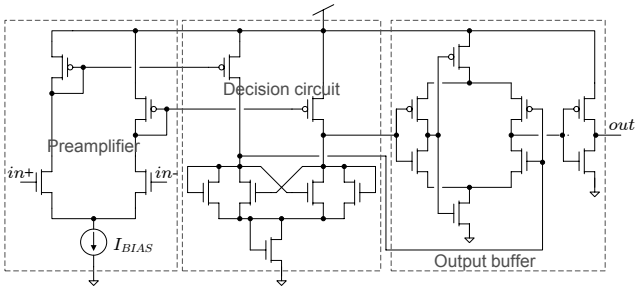


Figure 5: High-speed, low offset comparator [11].

The charge pump is implemented according to the design in [12]. As can be seen in Fig. 6, individual control of the increase and decrease rate is made possible by separate biasing of the source, and sink currents in the charge pump, through V_{bp} and V_{bn} respectively.

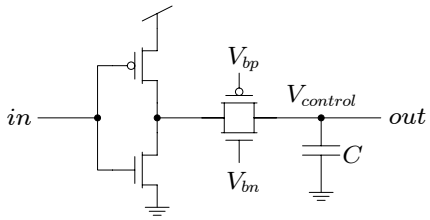


Figure 6: Charge pump with individual controls for the increase and decrease rate [12].

5. EXPERIMENTAL MEASUREMENTS

A prototype chip was fabricated to provide control of 8 variables. Control of more variables can be, however, achieved by putting in parallel multiple chips and applying copies of the cost function output to their inputs. A micrograph of the entire chip is shown in Fig. 7, where the floor plan with eight channels is delineated.

A printed circuit board was also designed and fabricated in order to test the chip (Fig. 8). The board features 16-bit DACs for accurate control of the biasing currents, high-speed buffers of 1.7GHz gain-bandwidth at the output of the

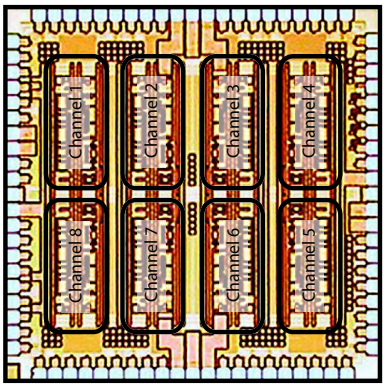


Figure 7: Micrograph of the entire chip. The chip dimensions are 3mm×3mm and it was implemented in a 0.5μm SiGe BiCMOS process.

channels and impedance matching to 50Ω at all input and outputs.

Figure 8: Printed Circuit Board used for chip characterization and testing.

Characterization of the main building blocks was performed before actual testing of the closed-loop performance. The range of achievable oscillation frequencies, generated by the 3-phase oscillator, was determined by setting I_{AMP} (Fig. 4(b)) to 0, which led to a desired maximum amplitude of 40-60mV$_{\text{pp}}$ for most of the frequencies. Fig. 10 illustrates the linear dependency of the oscillation frequency with the biasing current I_{FREQ}. An ultra wide tuning range for the oscillation frequency covering over 6 decades (from below 4kHz to above 600MHz) is demonstrated, making the system suitable for a wide variety of applications.

In order to demonstrate the synchronous detection performance of the chip, 4 channels were perturbed, each at a different frequency, from 98MHz to 158MHz and ∼20MHz apart, and their outputs were combined. Figure 9(a) shows the spectrum of the resulting multi-tone signal. The combined signal was fed back to the input of the chip and multiplied at each of the 4 channels with the corresponding dither frequency. Figure 9(b) illustrates the spectrum after multiplication with the 158MHz dither, but before any filtering, showing the expected products around 20MHz and

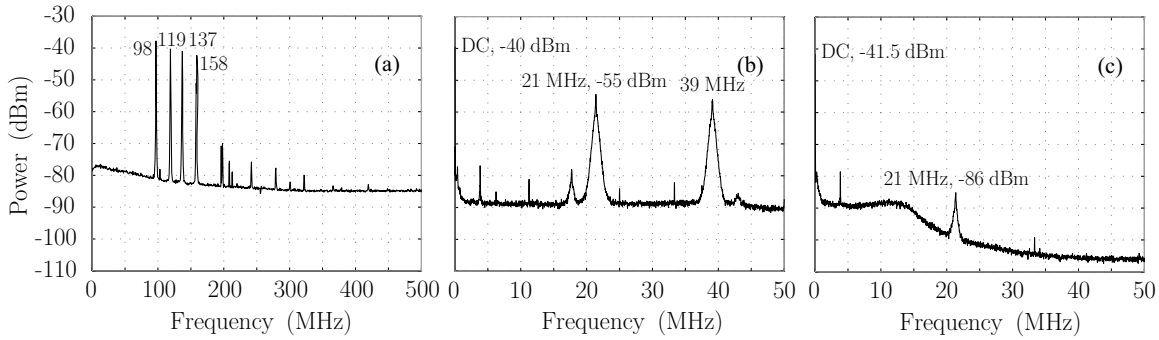


Figure 9: (a) Spectrum of a multi-tone signal with frequency components at 98MHz, 119MHz, 137MHz and 158MHz. The signal is injected at the input of the chip. (b) Spectrum after multiplication of the multi-tone signal with the 158MHz dither. (c) Spectrum after low-pass filtering the multiplier output (cut-off at 10MHz).

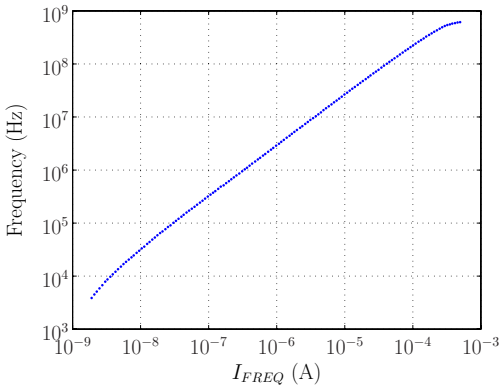
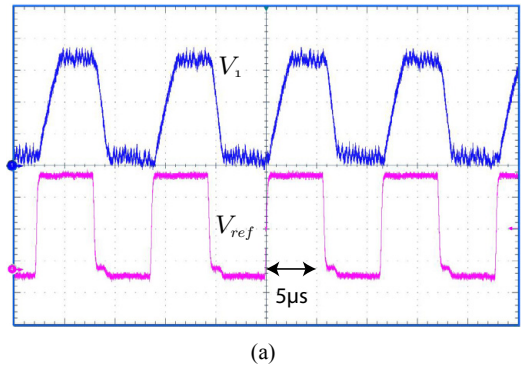


Figure 10: (a) Linear dependency of the dither frequency with current I_{FREQ}. Frequencies from below 4kHz to above 600MHz can be generated, adjusting one single control.

40MHz. Figure 9(c) displays the spectrum after applying the low-pass filter, set to a cut-off frequency of approximately 10MHz. Frequency components higher than the cut-off frequency are significantly attenuated and almost disappear below the noise floor. Comparison of Figs. 9(b) and (c) shows also the expected 30dB attenuation of the signal at 20MHz due to the 5^{th} order filter.

Finally, the closed-loop performance of the system was tested. A simple cost function using resistors and diodes, but with a wide operation bandwidth, was implemented according to the schematic of Fig. 11(b). The differential output $V_{max} - V_{min}$ realizes the function $f(V_1, \ldots, V_n, V_{ref}) = \max(V_1, \ldots, V_n, V_{ref}) - \min(V_1, \ldots, V_n, V_{ref}) - 2V_F$, where $V_i > 0,\ i = 1, \ldots, n$ are the voltage outputs from n channels of the system, $V_{ref} > 0$ is a reference voltage provided by a function generator, and V_F is the forward voltage drop of the used diodes. Function f has a global minimum, reached when $V_1 = \ldots = V_n = V_{ref}$. Figure 11(a) shows how the output from 1 channel (V_1) adapts to a 100kHz reference square wave V_{ref}, minimizing the output of the cost function. The dither was set at 20MHz. Adaptation in less than 3µs is observed.



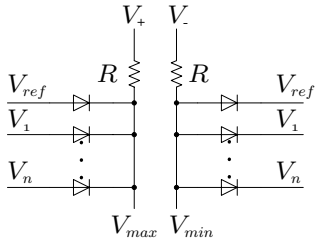


Figure 11: (a) Adaptation of 1 channel V_1 to a 100kHz reference square wave. Dither set at 20MHz. (b) Custom cost function using diodes and resistors.

6. CONCLUSION

The VLSI implementation of a model-free architecture for adaptive control, using the gradient descent flow algorithm and deterministic dithers for gradient estimation, has been presented. The circuit design of the main building blocks has been analyzed and the need for a fully translinear implementation has been elucidated. Characterization of the controller circuit components was performed and experimental results from closed-loop operation using a custom metric were demonstrated.

7. ACKNOWLEDGMENT

The authors would like to acknowledge the MOSIS Educational Program for providing the fabrication of the chip. Special thanks go also to Ralf Philipp and Abdullah Çelik for their valuable help. Integration of the chip with adaptive optics is currently pursued in collaboration with Dr. M. Vorontsov and supported by the Army Research Laboratory, Adelphi MD.

8. REFERENCES

[1] J. H. Winters, "Smart antennas for wireless systems," *IEEE Personal Commun. Mag.*, vol. 5, no. 1, pp. 23–27, Feb. 1998.

[2] T. Weyrauch and M. Vorontsov, "Free-space laser communications with adaptive optics: Atmospheric compensation experiments," *J. Opt. Fiber Commun. Rep.*, vol. 1, no. 4, pp. 355–379, Dec. 2004.

[3] A. Dembo and T. Kailath, "Model-free distributed learning," *IEEE Trans. Neural Networks*, vol. 1, no. 1, pp. 58–70, Mar. 1990.

[4] G. Cauwenberghs, "A fast stochastic error-descent algorithm for supervised learning and optimization," in *Adv. Neural Information Processing Systems (NIPS '92)*, vol. 5, San Mateo, CA: Morgan Kaufman, 1993, pp. 244–251.

[5] J. Alspector, R. Meir, B. Yuhas, A. Jayakumar, and D. Lippe, "A parallel gradient descent method for learning in analog VLSI neural networks," in *Adv. Neural Information Processing Systems (NIPS '92)*, vol. 5, San Mateo, CA: Morgan Kaufman, 1993, pp. 836–844.

[6] D. B. Kirk, D. Kerns, K. Fleischer, and A. H. Barr, "Analog VLSI implementation of multi-dimensional gradient descent," in *Adv. Neural Information Processing Systems (NIPS '92)*, vol. 5, San Mateo, CA: Morgan Kaufman, 1993, pp. 789–796.

[7] G. Cauwenberghs, "An analog VLSI recurrent neural network learning a continuous-time trajectory," *IEEE Trans. Neural Networks*, vol. 7, no. 2, pp. 346–361, Mar. 1996.

[8] D. N. Loizos, P. P. Sotiriadis, and G. Cauwenberghs, "A robust continuous-time multi-dithering technique for laser communications using adaptive optics," in *Proc. Int. Symp. Circuits and Systems (ISCAS '06)*, May 2006, pp. 3626–3629.

[9] ——, "Multi-channel coherent detection for delay-insensitive model-free adaptive control," in *Proc. Int. Symp. Circuits and Systems (ISCAS '07)*, May 2007.

[10] H. K. Khalil, *Nonlinear Systems*. MacMillan Publishing Company, 1992.

[11] R. J. Baker, H. W. Li, and D. E. Boyce, *CMOS Circuit Design, Layout and Simulation*. Prentice Hall, 1998.

[12] G. Cauwenberghs, "Analog VLSI stochastic perturbative learning architectures," *Int. J. Analog Integrated Circuits and Signal Processing*, vol. 13, no. 1-2, pp. 195–209, May-June 1997.