Synthesis of Bias-Scalable CMOS Analog Computational Circuits Using Margin Propagation

Ming Gu, *Student Member, IEEE*, and Shantanu Chakrabartty, *Senior Member, IEEE*

***Abstract*—Approximation techniques are useful for implementing pattern recognizers, communication decoders and sensory processing algorithms where computational precision is not critical to achieve the desired system level performance. In our previous work, we had proposed margin propagation (MP) as an efficient piecewise linear (PWL) approximation technique to a log-sum-exp function and had demonstrated its advantages for implementing probabilistic decoders. In this paper, we present a systematic and a generalized approach for synthesizing analog piecewise-linear (PWL) computing circuits using the MP principle. MP circuits use only addition, subtraction and threshold operations and hence can be implemented using universal conservation principles like the Kirchoff's current law. Thus, unlike the conventional translinear CMOS current-mode circuits, the operation of the MP circuits are functionally similar in weak, moderate, and strong inversion regimes of the MOS transistor making the design approach bias-scalable. This paper presents measured results from MP circuits prototyped in a 0.5 μm standard CMOS process verifying the bias-scalable property. As an example, we apply the synthesis approach towards designing linear classifiers and we verify its performance using measured results.**

***Index Terms*—Analog computation, current-mode circuits, margin propagation (MP), moderate-inversion circuits, piecewise-linear (PWL) circuit, translinear.**

I. Introduction

FOR applications that do not require high computational precision (e.g., pattern matching), current-mode analog signal processing (ASP) offers several advantages over digital signal processing techniques in terms of energy efficiency and speed [1], [2]. By exploiting the computational primitives inherent in the device physics, complicated mathematical functions can be implemented in analog, using significantly lower number of transistors as compared to its digital counterparts [2], [3]. The most popular approach for synthesizing current-mode analog computational circuits is based on the translinear principle which exploits the exponential current-to-voltage relationship observed in bipolar transistors [4], [5] and metal-oxide-semiconductor (MOS) transistors biased in weak-inversion [6]–[8]. In [9] and [10], the translinear principle has also been extended towards synthesizing current-mode circuits using MOS transistors biased in strong-inversion.

Manuscript received February 24, 2011; revised May 31, 2011; accepted July 08, 2011. Date of publication September 19, 2011; date of current version January 27, 2012. This work was supported by the National Science Foundation (NSF) under Grant CCF:0728996. This paper was recommended by Associate Editor V. Gaudet.

The authors are with the Department of Electrical and Computer Engineering, Michigan State University (e-mail: shantanu@egr.msu.edu).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TCSI.2011.2163968

All CMOS based translinear synthesis techniques share one common property: they rely on the physical response of the MOS transistor operating either in strong or weak inversion. Hence, translinear circuits designed to operate in one biasing regime are generally not portable to the other. Also, while operating in weak inversion is beneficial for achieving superior energy efficiency and while operating in strong inversion is beneficial for achieving higher speed, it has been argued that a better speed energy trade-off is achievable when the transistors are biased in the moderate inversion regime [11], [12]. Currently, no current-mode synthesis technique exists for implementing mathematical functions using MOS transistors biased in moderate inversion. However, if we can design circuits that operate using only universal conservation laws (e.g., current or charge conservation), the circuits should be portable across different biasing conditions.

In this paper we propose a bias-scalable framework for synthesizing analog computational circuits based on a margin propagation (MP) principle which uses only thresholding operations and the Kirchoff's current conservation principle. MP principle was introduced in [13] as a piecewise linear (PWL) approximation to the log-sum-exp function which is extensively used in communications [14], optimization [15], and learning [13] algorithms. In this paper we show that the MP principle can be used to implement a wide-range of linear and non-linear mathematical functions. An important attribute of MP based synthesis is that all computations are performed in the logarithmic domain, as a result of which multiplications map into additions, divisions map into subtractions, and power computations map into multiplications. This simplifies the implementation of many complicated non-linear functions.

We illustrate the basic synthesis procedure using a simple example. Consider a function f given by

$$f = a_1 \cdot b_1 + a_2 \cdot b_2 \tag{1}$$

where $a_1, a_2, b_1, b_2 \in \mathbb{R}$ are the operands and $\cdot$ denotes a multiplication operator. The operands can be converted into their differential form according to

$$a_1 = a_1^+ - a_1^-, \quad a_2 = a_2^+ - a_2^-,$$
$$b_1 = b_1^+ - b_1^-, \quad b_2 = b_2^+ - b_2^-,$$

where $a_1^+, a_1^-, a_2^+, a_2^-, b_1^+, b_1^-, b_2^+, b_2^- > 0$ are positive differential components. Each of these positive components can be converted into its logarithmic form as

$$L_{a_1}^+ = \log a_1^+, \quad L_{a_1}^- = \log a_1^-,$$
$$L_{a_2}^+ = \log a_2^+, \quad L_{a_2}^- = \log a_2^-,$$
$$L_{b_1}^+ = \log b_1^+, \quad L_{b_1}^- = \log b_1^-,$$
$$L_{b_2}^+ = \log b_2^+, \quad L_{b_2}^- = \log b_2^-.$$

1549-8328/$26.00 © 2011 IEEE

Authorized licensed use limited to: J.R.D. Tata Memorial Library Indian Institute of Science Bengaluru. Downloaded on December 04,2024 at 10:41:39 UTC from IEEE Xplore. Restrictions apply.

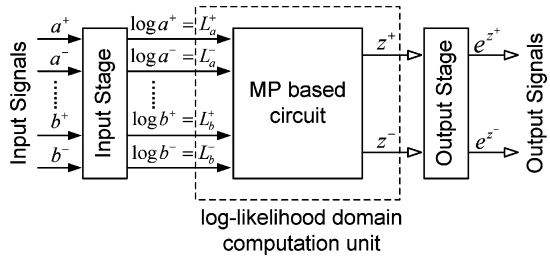


Fig. 1. Generic architecture of an MP-based analog processor.

Accordingly, (1) can be converted into

$$f = e^{z^+} - e^{z^-}, \tag{2}$$

where z^+ and z^- are expressed as

$$\begin{aligned} z^+ &= \log\left(e^{L_{a_1}^+ + L_{b_1}^+} + e^{L_{a_1}^- + L_{b_1}^-} + e^{L_{a_2}^+ + L_{b_2}^+} + e^{L_{a_2}^- + L_{b_2}^-}\right) \\ z^- &= \log\left(e^{L_{a_1}^+ + L_{b_1}^-} + e^{L_{a_1}^- + L_{b_1}^+} + e^{L_{a_2}^+ + L_{b_2}^-} + e^{L_{a_2}^+ + L_{b_2}^-}\right). \end{aligned} \tag{3}$$

Thus, evaluation of the function f can be decomposed into three steps: a) input processing step which computes the log-likelihoods L_x^+ and L_x^- based on the input x; b) log-sum-exp computation step given by (3); and c) output processing step which computes f based on (2). We will show in this paper that for most functions, the input and the output processing steps are similar to equations shown above. The only difference in processing lies in the form of the log-sum-exp functions as shown in (3).

In MP based synthesis the objective will be to approximate the log-sum-exp functions and in the process generate different forms of linear and non-linear functions. Fig. 1 shows the architecture of a generic MP-based analog signal processor. The input differential signals $a = a^+ - a^-$, $b = b^+ - b^-$ are converted into their logarithmic form, processed by an MP circuit and the differential outputs z^+ and z^- are used to compute the final output according to (2). For many applications, for instance in pattern recognition, only the sign of $z^+ - z^-$ is important, so the output stage could potentially be replaced by just a comparator.

The paper is organized as follows: Section II briefly describes the mathematical concepts underlying the MP based approximation, followed by Section III, which presents examples of MP circuits implementing the basic mathematical functions. We juxtapose the theoretical and measured results in this section to demonstrate the bias-scalable property of the MP circuits. All the measured results presented in this paper have been obtained from circuits prototyped in a standard 0.5 μm CMOS technology. Section IV analyzes the scalability of MP circuits with respect to fundamental noise, transistor mismatch and speed. In Section V, we illustrate the synthesis procedure by designing an MP based linear pattern classifier. In Section VI, we conclude the paper by describing some extensions of the proposed work.

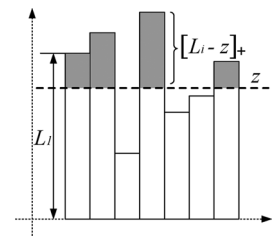


Fig. 2. Illustration of reverse water-filling procedure.

II. Log-Sum-Exp Approximation Using Margin Propagation

Given a set of likelihood scores L_i, $i = 1, \ldots, N$, a general form of log-sum-exp function is given by

$$z_{\log} = \log\left(\sum_{i=1}^{N} e^{L_i}\right). \tag{4}$$

In MP approximation, $z_{\log}$ is estimated using z, which is computed according to a reverse water-filling constraint [16] expressed as

$$\sum_{i=1}^{N} [L_i - z]_+ = \gamma, \tag{5}$$

where $[\cdot]_+ = \max(\cdot, 0)$ denotes a threshold operation and $\gamma \geq 0$ represents a parameter of the algorithm. The solution to (5) can be visualized using Fig. 2, where the cumulative score beyond z (shown by the shaded area) equals γ. It can be shown that z is a PWL approximation to $z_{\log}$ and the sketch of the proof is summarized in Appendix I. For the rest of this paper, we will denote the MP function as

$$z = M(L_1, \ldots, L_N, \gamma). \tag{6}$$

The MP based PWL approximation is illustrated graphically for the following function:

$$\phi(L_1, L_2) = \log\left(\frac{1 + e^{L_1 + L_2}}{e^{L_1} + e^{L_2}}\right), \tag{7}$$

which is commonly used for implementing communication decoders [17], [18]. First, similar to our example in Section I, we express $\phi(\cdot)$ in terms of the differential components of its arguments, i.e., $L_1 = L_1^+ - L_1^-$, $L_2 = L_2^+ - L_2^-$ as

$$\begin{aligned} \phi(L_1, L_2) &= \log\left(\frac{e^{L_1^- + L_2^-} + e^{L_1^+ + L_2^+}}{e^{L_1^- + L_2^+} + e^{L_1^+ + L_2^-}}\right), \\ &= \log\left(e^{L_1^- + L_2^-} + e^{L_1^+ + L_2^+}\right) \\ &\quad - \log\left(e^{L_1^- + L_2^+} + e^{L_1^+ + L_2^-}\right). \end{aligned} \tag{8}$$

Thus, $\phi(L_1, L_2)$ is represented as a difference of two "log-sum-exp" functions, each of which is approximated by its MP equivalent as

$$\begin{aligned} \phi(L_1, L_2) \approx\ & M(L_1^- + L_2^-, L_1^+ + L_2^+, \gamma) \\ & - M(L_1^+ + L_2^-, L_1^- + L_2^+, \gamma). \end{aligned} \tag{9}$$

Authorized licensed use limited to: J.R.D. Tata Memorial Library Indian Institute of Science Bengaluru. Downloaded on December 04,2024 at 10:41:39 UTC from IEEE Xplore. Restrictions apply

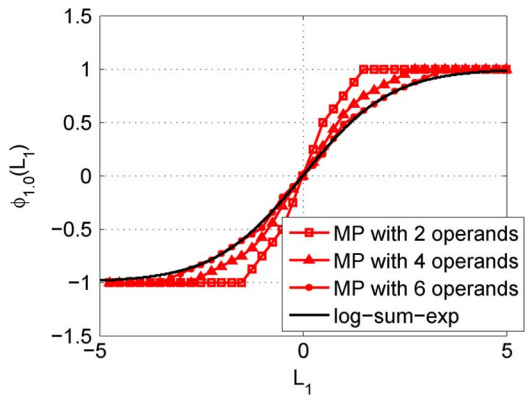


Fig. 3. Trade-off between quality of approximation and the number of operands as achieved by MP function.

Fig. 3 shows the MP-based PWL approximation of $\phi(\cdot)$ for a fixed value of γ, when one of the operands L_1 is varied. The figure also shows the results when different number of operands $L_1, \ldots, L_N$ are used in (8). The results visually show that as the number of operands increase, more linear splines are introduced in the MP function which improves the quality of the approximation. This trade-off between the approximation error and the number of operands also holds for other forms of the log-sum-exp functions. In [19] we used the MP approximation to implement an analog LDPC decoder and compared its performance with an equivalent decoder [20] using a min-sum approximation of (7). We would like to point out that MP approximation is more general than the min-sum approximation and both approximations are identical when $\gamma \to 0$.

III. MP Based Computational Circuits

A. Basic MP Circuit

A current-mode circuit implementing the basic reverse water-filling equation (5) is shown in Fig. 4. The four operands $L_1 \sim L_4$ in (6) are represented by currents $I_{L_1} \sim I_{L_4}$. The hyper-parameter γ in (6) is implemented by the current I_γ. The circuit in Fig. 4 bears similarity to other parallel current-mode circuits like the winner-take-all circuits. However, as mentioned in the previous section, the MP implementation is more general and for $I_\gamma \to 0$ the MP circuit implements a winner-take-all function. The Kirchoff's current law (KCL) when applied to node A is equivalent to the reverse water-filling condition $\sum_{i=1}^{4}[I_{L_i} - I_z]_+ = I_\gamma$. I_z represents the MP approximation $M(L_1, \ldots, L_4, \gamma)$, and $[\cdot]_+$ denotes a rectify operation which is implemented by the PMOS diodes $P_1 \sim P_4$. It can be readily verified that the PMOS diodes ensure that $V_{\mathrm{DS},N_i} = V_{\mathrm{SG},P_i} + V_{\mathrm{DS,sat}}$, where V_{DS,N_i} is the drain-to-source voltage drop across transistor $N_i, i = 1, \ldots, 4$, V_{SG,P_i} is the voltage drop across the PMOS diodes, and $V_{\mathrm{DS,sat}}$ is the saturation voltage across the current sink I_γ. Thus, transistors $N_1 \sim N_4$ are always maintained in saturation irrespective of the operation region (weak, moderate, and strong inversion). This is important because the operation of the MP circuit relies on the accuracy of the current mirrors formed by $N_0 \sim N_4$. Also, to ensure proper

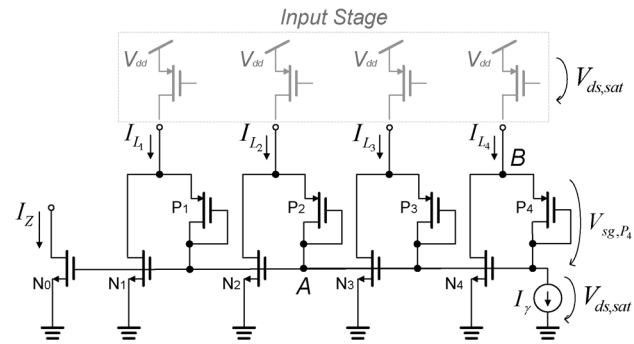


Fig. 4. CMOS implementation of the core MP circuit.

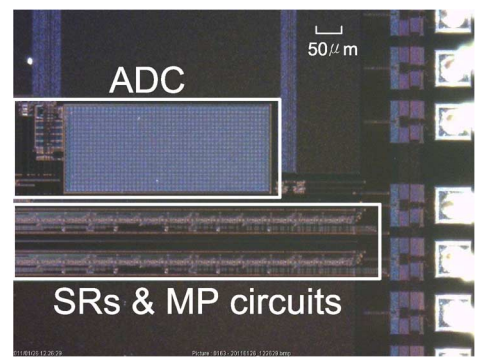


Fig. 5. Die micrograph of the chip.

operation of the core MP circuit the input currents have to satisfy the constraint $\sum_{i=1}^{4} I_{L_i} \geq I_\gamma$ and the input current range can be expressed as

$$\frac{1}{2}\mu_n C_{\mathrm{ox}} \left(\frac{W}{L}\right) \left(\frac{1}{2}V_{dd}\right)^2 \geq I_{L_i} \geq 0. \tag{10}$$

In the derivation of the upper-bound of the input current range we have assumed that the $I_{L_i} \gg I_\gamma$ and the PMOS and the NMOS transistors have similar drain-to-source voltage drops (implying $V_B \approx V_{dd}/2$). It should be noted that the PMOS diodes in circuit shown in Fig. 4 introduces a threshold voltage drop, which implies that the supply voltage has to be greater than $2V_{\mathrm{GS}} + V_{\mathrm{DS}}$. However, a low-voltage implementation of the circuit in Fig. 4 is possible by replacing the PMOS diodes by low-threshold active diodes [21], but at the expense of larger silicon area, higher power dissipation, and degraded speed and stability.

We have prototyped a programmable version of the MP circuit shown in Fig. 4 in a 0.5 μm standard CMOS process. The architecture of the prototype is shown in Fig. 6 and its micrograph is shown in Fig. 5. A serial shift-register is used for selectively turning on and off each of different branches of currents. In this manner, the same MP circuit in Fig. 4 can be reconfigured to implement different mathematical functions. The architecture in Fig. 6 uses an on-chip first-order $\Sigma\Delta$ modulator as an analog-to-digital converter for measurement and calibration of the output currents. Because the circuits used for implementing the $\Sigma\Delta$ modulator have been reported elsewhere [22] we have omitted its implementation details in this paper. The input currents and hence the operating region (weak, moderate, and strong inversion) of all the transistors are adjusted by controlling the gate voltages of the PMOS transistors. For each of

Authorized licensed use limited to: J.R.D. Tata Memorial Library Indian Institute of Science Bengaluru. Downloaded on December 04,2024 at 10:41:39 UTC from IEEE Xplore. Restrictions apply

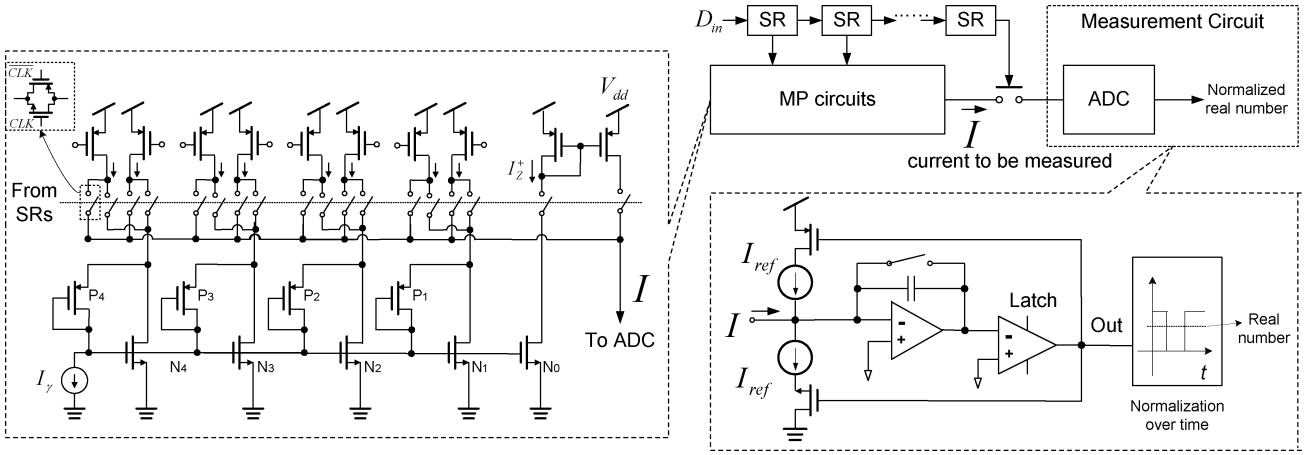


Fig. 6. System architecture of the chip.

the operating regions, the ADC is recalibrated to the maximum current and the measured results (presented in the next sections) are normalized with respect to the maximum current.

B. MP Based Addition

To compute $a+b$ with respect to the two input variables a and b, $a \in \mathbb{R}, b \in \mathbb{R}$ the operands are first represented in their differential forms $a = a^+ - a^-$ and $b = b^+ - b^-$ with $a^+, a^-, b^+, b^- \in \mathbb{R}^+$. Then, the operation is rewritten as in Fig. 1

$$\begin{aligned} a+b &= a^+ - a^- + b^+ - b^- \\ &= (a^+ + b^+) - (a^- + b^-) \\ &= e^{\log(a^+ + b^+)} - e^{\log(a^- + b^-)}. \end{aligned} \tag{11}$$

All the operands are then mapped into a log-likelihood domain according to

$$\begin{aligned} L_a^+ &= \log a^+, \quad L_a^- = \log a^-, \\ L_b^+ &= \log b^+, \quad L_b^- = \log b^-, \end{aligned}$$

where $L_a^+ \in \mathbb{R}$, $L_a^- \in \mathbb{R}$, $L_b^+ \in \mathbb{R}$, $L_b^- \in \mathbb{R}$. Since our circuit implementation uses only unipolar current sources, we impose additional constraints $a^+, a^-, b^+, b^- > 1$. Due to the differential representation, this additional constraint will not affect the final output. Let z^+ denote $\log(a^+ + b^+)$ and z^- denote $\log(a^- + b^-)$. Then, (11) can be written as

$$a + b = e^{z^+} - e^{z^-} \tag{12}$$

and z^+ can be expanded as

$$\begin{aligned} z^+ &= \log(a^+ + b^+) \\ &= \log\left(e^{\log a^+} + e^{\log b^+}\right) \\ &= \log\left(e^{L_a^+} + e^{L_b^+}\right). \end{aligned} \tag{13}$$

We now apply the MP approximation to the log-sum-exp functions to obtain

$$z^+ \approx M(L_a^+, L_b^+, \gamma). \tag{14}$$

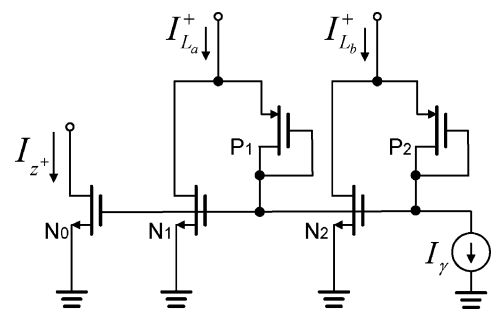


Fig. 7. MP circuit for implementing two operand addition.

Similarly,

$$\begin{aligned} z^- &= \log(a^- + b^-) \\ &= \log\left(e^{\log a^-} + e^{\log b^-}\right) \\ &= \log\left(e^{L_a^-} + e^{L_b^-}\right) \\ &\approx M(L_a^-, L_b^-, \gamma). \end{aligned} \tag{15}$$

Since the (14) and the (15) uses only two operands, its circuit level implementation is identical to Fig. 4 except that it has only two input branches instead of four. The single-ended circuit for addition is shown in Fig. 7. I_{z^+}, the current through N_0, represents the value of z^+. $I_{L_a^+}$, $I_{L_b^+}$, and I_γ represents L_a^+, L_b^+ and γ respectively.

Fig. 8(a)–(c) compares the output $z^+ - z^-$ obtained using floating-point implementation of the "log-sum-exp" functions, using floating-point implementation of the MP approximation and from measurements (normalized currents) using the fabricated prototype.

Fig. 9(a)–(c) show the measurement results when the transistors are biased in (a) strong inversion; (b) moderate inversion; and (c) weak inversion. Fig. 9(d)–(f) show the approximation error between the measured response and the software model. The results validate our claim for the addition operation that MP approximation is scalable across different biasing regions and the circuit in Fig. 7 approximates the "log-sum-exp" function with a reasonable error. The error increases when the circuit is biased in the weak inversion and as with any subthreshold analog VLSI circuit, which we attribute to transistor mismatch,

Authorized licensed use limited to: J.R.D. Tata Memorial Library Indian Institute of Science Bengaluru. Downloaded on December 04,2024 at 10:41:39 UTC from IEEE Xplore. Restrictions apply

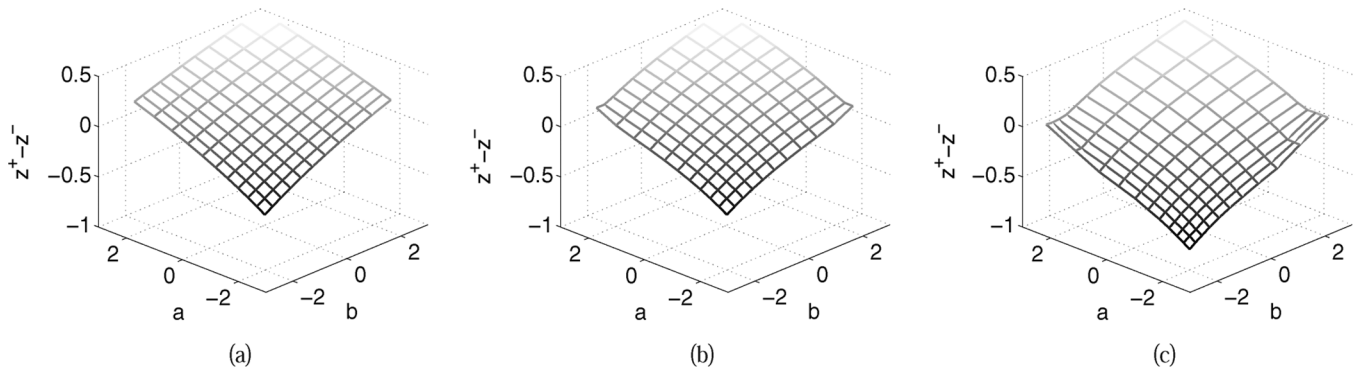


Fig. 8. Addition: $z^+ - z^-$ computed according to (a) the original log-sum-exp function; (b) MP approximations (simulation); and (c) MP circuits (measurement).

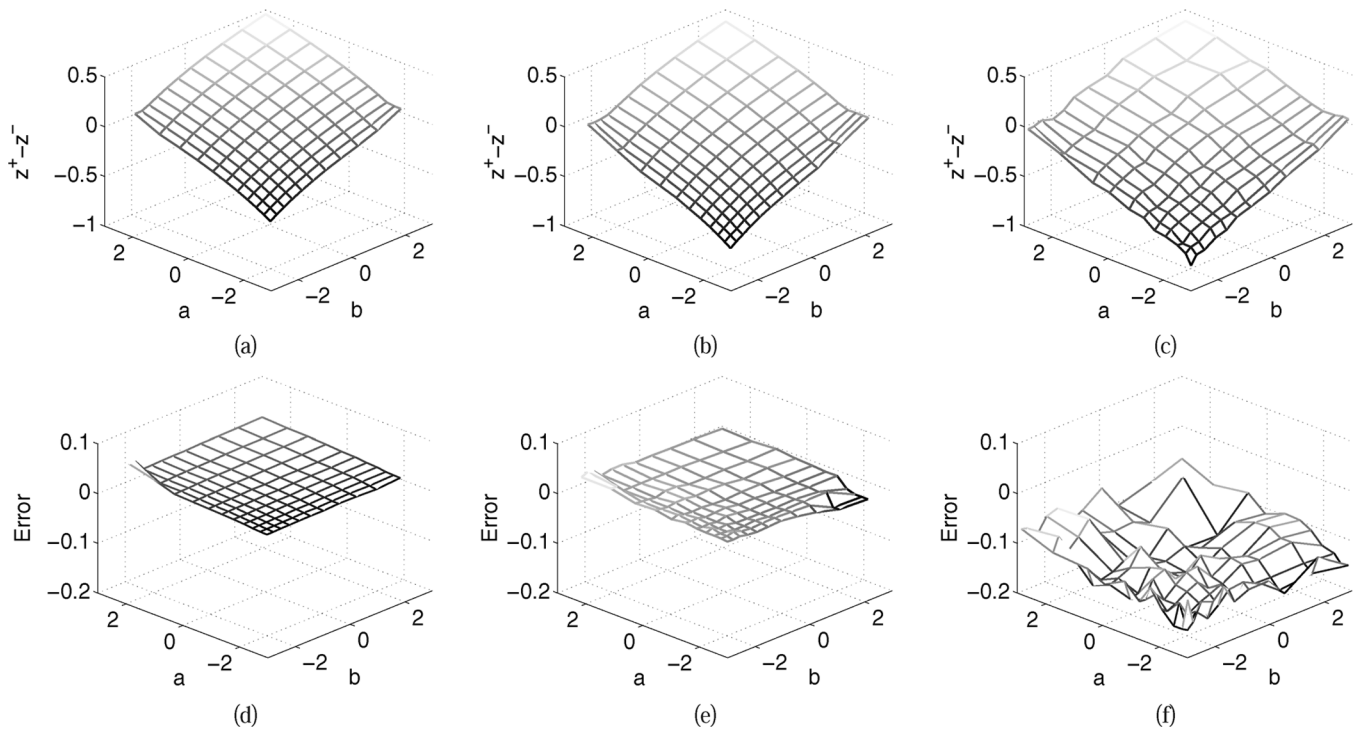


Fig. 9. Addition: measurement results for $z^+ - z^-$ with transistors biased in (a) strong inversion; (b) moderate inversion; (c) weak inversion; (d)–(f) computation errors for (a)–(c).

flicker-noise, and errors introduced in mapping floating-point operands into analog operands (currents).

C. MP Based Subtraction

Synthesizing an MP circuit for subtraction $a-b$, with a and b being the two operands is similar to the addition operation. The operation can be written in a differential form as $a = a^+ - a^-$ and $b = b^+ - b^-$, with $a^+, a^-, b^+, b^- \in \mathbb{R}^+$ and satisfying $a^+, a^-, b^+, b^- > 1$. Thus, $a - b$ is written as

$$\begin{aligned} a - b &= (a^+ - a^-) - (b^+ - b^-) \\ &= (a^+ + b^-) - (a^- + b^+) \\ &= e^{\log(a^+ + b^-)} - e^{\log(a^- + b^+)} \end{aligned} \tag{16}$$

which after conversion into a log-likelihood domain

$$\begin{aligned} L_a^+ &= \log a^+, \quad L_a^- = \log a^-, \\ L_b^+ &= \log b^+, \quad L_b^- = \log b^-, \end{aligned}$$

leads to

$$\begin{aligned} z^+ &= \log(a^+ + b^-) \\ &= \log\left(e^{\log a^+} + e^{\log b^-}\right) \\ &= \log\left(e^{L_a^+} + e^{L_b^-}\right) \\ &\approx M(L_a^+, L_b^-, \gamma) \end{aligned} \tag{17}$$

and

$$\begin{aligned} z^- &= \log(a^- + b^+) \\ &= \log\left(e^{\log a^-} + e^{\log b^+}\right) \\ &= \log\left(e^{L_a^-} + e^{L_b^+}\right) \\ &\approx M(L_a^-, L_b^+, \gamma) \end{aligned} \tag{18}$$

where z^+ and z^- are differential forms of the output log-likelihoods as described in (2). The circuit implementation of subtraction is identical to that of addition, except that the input differential currents are now permuted. Fig. 10 compares the

Authorized licensed use limited to: J.R.D. Tata Memorial Library Indian Institute of Science Bengaluru. Downloaded on December 04,2024 at 10:41:39 UTC from IEEE Xplore. Restrictions apply

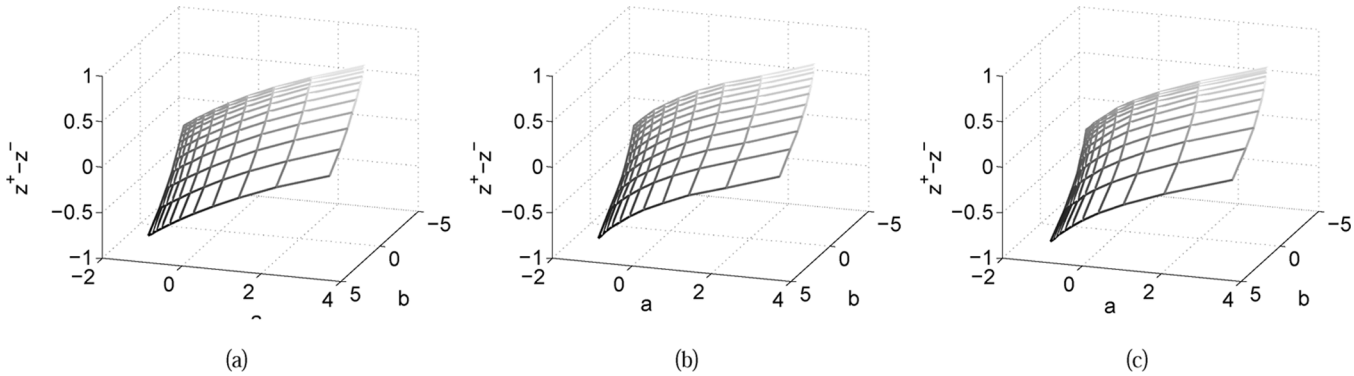


Fig. 10. Subtraction: $z^+ - z^-$ computed by (a) original functions; (b) MP approximations (simulation); and (c) MP circuits (measurement).

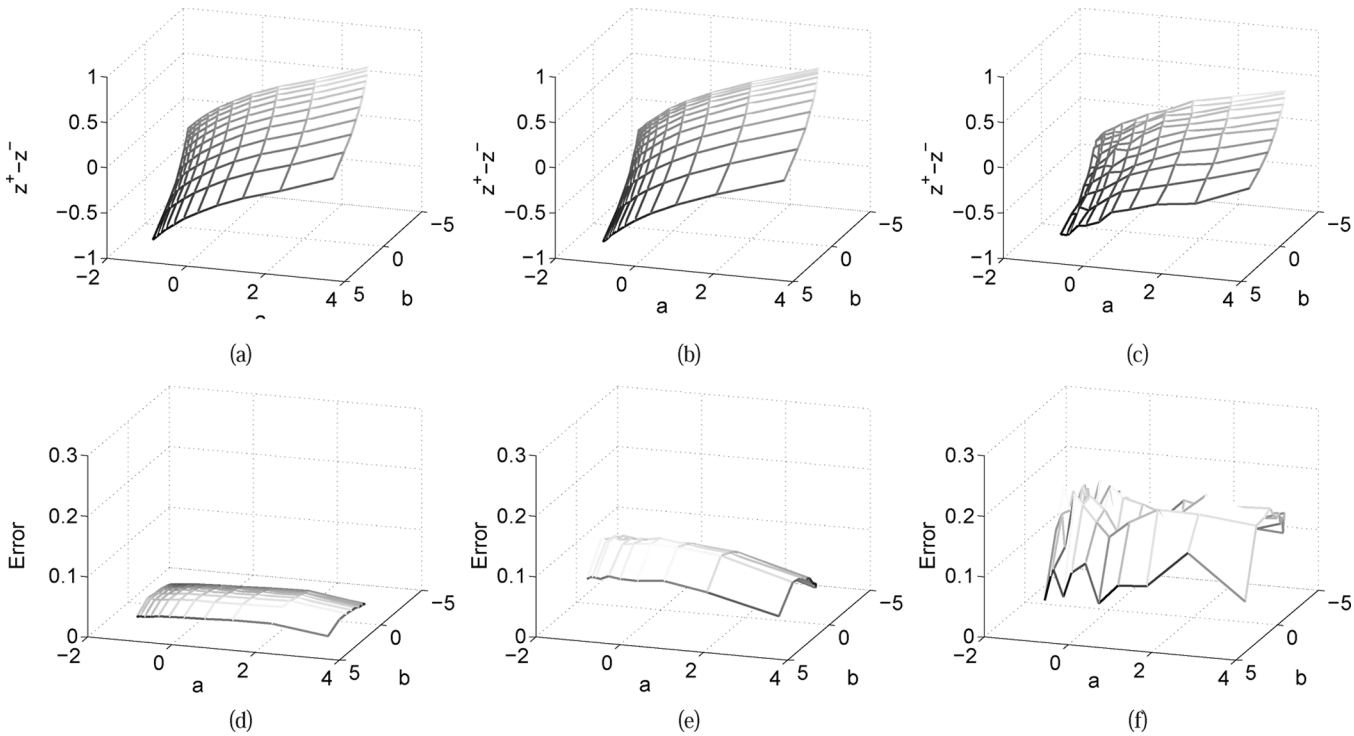


Fig. 11. Subtraction: measurement results for $z^+ - z^-$ with transistors biased in (a) strong inversion; (b) moderate inversion; (c) weak inversion; and (d)–(f) show respective approximation errors.

measured response with software based "log-sum-exp" and MP implementation. Also, Fig. 11(a)–(c) shows the measured results when the transistors are biased in the three different operating regimes. Similar to the results obtained for the addition operation, the measured result show the bias-scalable operation of the MP circuits.

D. Four Quadrant Multiplication

Implementing a single quadrant multiplier using MP circuit is straight-forward because $z = a \cdot b$ in its logarithmic form can be expressed as a sum of likelihoods $L_z = L_a + L_b$, where L_x denotes the logarithm of the operand x. Thus, a single quadrant multiplier requires only a current summation and can be implemented using KCL. For an MP-based four-quadrant multiplier the operation $a \cdot b$ is again expressed in terms of the differential operands as

$$\begin{aligned} a \cdot b &= (a^+ - a^-) \cdot (b^+ - b^-) \\ &= (a^+ \cdot b^+ + a^- \cdot b^-) - (a^- \cdot b^+ + a^+ \cdot b^-) \\ &= e^{\log(a^+ \cdot b^+ + a^- \cdot b^-)} - e^{\log(a^- \cdot b^+ + a^+ \cdot b^-)}. \end{aligned} \tag{19}$$

After log-likelihood mapping, the following differential forms are obtained:

$$\begin{aligned} z^+ &= \log(a^+ \cdot b^+ + a^- \cdot b^-) \\ &= \log\left(e^{\log(a^+ \cdot b^+)} + e^{\log(a^- \cdot b^-)}\right) \\ &\approx M(L_a^+ + L_b^+, L_a^- + L_b^-, \gamma) \end{aligned} \tag{20}$$

and

$$\begin{aligned} z^- &= \log(a^- \cdot b^+ + a^+ \cdot b^-) \\ &= \log\left(e^{\log(a^- \cdot b^+)} + e^{\log(a^+ \cdot b^-)}\right) \\ &\approx M(L_a^- + L_b^+, L_a^+ + L_b^-, \gamma). \end{aligned} \tag{21}$$

The circuit implementing an MP based four-quadrant multiplier is shown in Fig. 12. I_{z^+} denotes the value of z^+. I_γ represents γ. $I_{L_a^+}(I_{L_a^-}), I_{L_b^+}(I_{L_b^+})$ represent $L_a^+(L_a^-), L_b^+(L_b^-)$ respectively.

Fig. 13 compares the measured output $z^+ - z^-$ with the ideal implementation of the model (20) and (21). Fig. 14(a)–(c) shows the measurement results when the transistors are biased in: (a)

Authorized licensed use limited to: J.R.D. Tata Memorial Library Indian Institute of Science Bengaluru. Downloaded on December 04,2024 at 10:41:39 UTC from IEEE Xplore. Restrictions apply

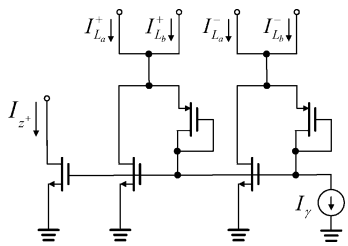


Fig. 12. MP circuit for implementing four quadrant operand multiplication.

strong; (b) moderate; and (c) weak inversion along with the respective approximation error. Again, the measured results show the bias-scaling property of MP circuits.

E. Single Quadrant Division

Like the implementation of a single quadrant multiplication, implementing a single quadrant division in logarithmic domain is straightforward. For operands $a, b > 0$, division in the logarithmic domain is equivalent to a subtraction operation $L_a - L_b$ which can be implemented using KCL. To implement four quadrant division, the operation first requires extracting the sign of the operand as $x = \mathrm{sgn}(x) \cdot |x|$, where $|x|$ is the absolute value of x. A single quadrant division can now be used for the absolute values and the sign of the final output can be computed using only comparators.

F. Power and Polynomial Computation

For positive operands satisfying $a \in \mathbb{R}^+$, $a > 1$, computing a^n, $n \in \mathbb{Z}^+$ in the logarithmic domain is equivalent to

$$\begin{aligned}\log(a^n) &= n \cdot \log(a),\\ &= n \cdot L_a \end{aligned} \tag{22}$$

which can be implemented using a simple current mirror as shown in Fig. 15. Thus, in MP-based synthesis, power functions like square-root or cube-root can be implemented using current mirrors which significantly simplifies the implementation compared to translinear based synthesis [7]. However, for a four-quadrant implementation of power and polynomial function would require evaluating a Taylor expansion using multiplication and addition operations. Details of the implementation is provided in Appendix II.

As an example, a four quadrant square function a^2 can be implemented by expressing it as a product of two differential operands as $(a^+ - a^-) \cdot (a^+ - a^-)$, which has an architecture similar to the four-quadrant multiplier described in Section III-D. The differential output z^+ and z^- corresponding to the square operation can be expressed in terms of differential log-likelihood inputs L_a^+, L_a^- as

$$z^+ \approx M(2L_a^+, 2L_a^-, \gamma) \tag{23}$$

$$z^- = \log 2 + L_a^+ + L_a^- \tag{24}$$

Fig. 16 shows the measured results when the circuit in Fig. 12 has been configured to compute a four-quadrant square function. Again, the circuit is shown to approximate the "log-sum-exp" function for strong-inversion, moderate-inversion, and weak-inversion biasing conditions.

IV. Scalability Analysis of MP Circuit

In Fig. 3, we showed that the accuracy of the PWL approximation improves when the number of operands in the margin function increases. In this section, we analyze the effect of hardware artifacts (noise, mismatch, speed, and input/output impedance) on the performance and scalability of MP circuits.

A. Noise and Mismatch Analysis of MP Circuit

The noise model for the basic MP circuit in Fig. 4 is shown in Fig. 17. For the following analysis we will assume that the number of input current branches in the MP circuit equals K. Due to the MP constraint (5) we will assume that the noise in only $K-1$ of the parallel branches are uncorrelated (due to KCL constraints). The total output noise power $I_{n,z}^2$ can therefore be expressed as

$$\begin{aligned} I_{n,z}^2 = \sum_{i=1}^{K} \Bigg[&\frac{I_{n,\mathrm{source}_i}^2}{(K-2)^2} + \frac{I_{n,N_i}^2}{(K-2)^2} \\ &+ \frac{I_{n,P_i}^2}{g_m^2 R_{\mathrm{source}}^2 (K-2)^2} \Bigg] \\ &+ \frac{I_{n,\mathrm{sink}}^2}{K^2} + I_{n,N_0}^2 \end{aligned} \tag{25}$$

where I_{n,source_i}, I_{n,N_i}, I_{n,P_i}, and $I_{n,\mathrm{sink}}$ are noise currents as shown in Fig. 17 and R_{source} is the output impedance of the current sources. We have simplified the expression in (25) by assuming K to be large and by assuming that the small-signal parameters g_m, R_{source} for all the transistors are approximately equal. Equation (25) can be written as

$$I_{n,z,\mathrm{thermal}}^2 = \left[1 + \frac{2}{K} + \frac{1}{g_m^2 R_{\mathrm{source}}^2 K}\right] \frac{8}{3} kT g_m, \tag{26}$$

where we have considered only the effect of thermal noise as given by $I_n^2 = 8/3kTg_m$ [23]. The effect of the flicker-noise has been ignored in (26), because it exhibits a similar dependency on K as the thermal-noise. Equation (26) shows that as the number of branches (or operands) K increases, the output noise reduces and in the limit $K \to \infty$, the output noise is just equal to the noise due to transistor N_0. The reduction in the output noise with the increase in K can be attributed to the attenuation and averaging effect of the K stage current mirrors formed by $N_1 \sim N_K$. Also, it is important that $g_m R_{\mathrm{source}} \gg 1$, not only for reducing the effect of noise but also to ensure that the input impedance of the MP circuit is much lower than the output impedance of the current sources (see Fig. 17).

The mismatch analysis for the MP circuit follows a similar procedure as the noise analysis. The MOS transistor mismatch models used in the analysis are based on the work reported in [24], which uses two technology dependent constants A_{V_T} and A_β that determine the mismatch variance as

$$\sigma^2(\Delta V_T) = \frac{A_{V_T}^2}{WL} \tag{27}$$

$$\left(\frac{\sigma(\Delta\beta)}{\beta}\right)^2 = \frac{A_\beta^2}{WL}. \tag{28}$$

V_T denotes the threshold voltage, $\beta = \mu C_{\mathrm{ox}} W/L$, W, and L represent the width and length of the MOS transistor. It is

Authorized licensed use limited to: J.R.D. Tata Memorial Library Indian Institute of Science Bengaluru. Downloaded on December 04,2024 at 10:41:39 UTC from IEEE Xplore. Restrictions apply.

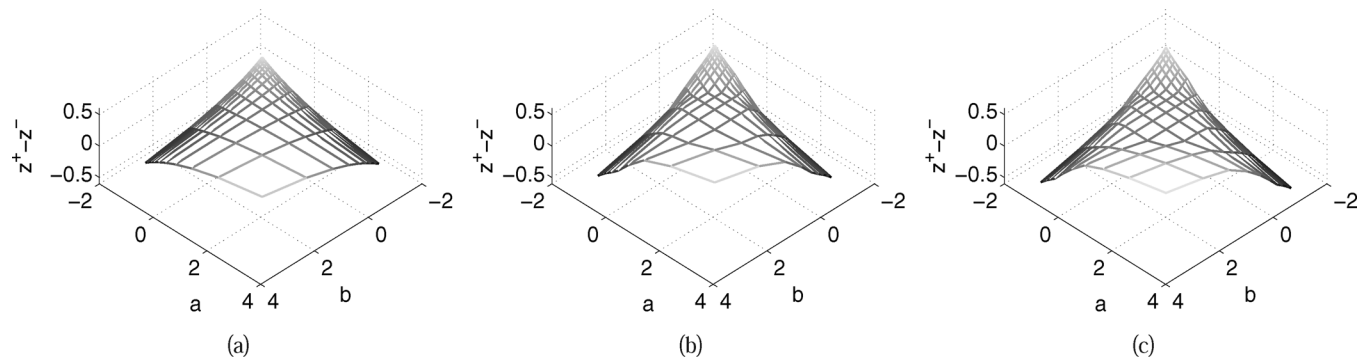


Fig. 13. Multiplication: $z^+ - z^-$ computed according to the (a) original function; (b) MP approximations (simulation); and (c) MP circuits (measurement).

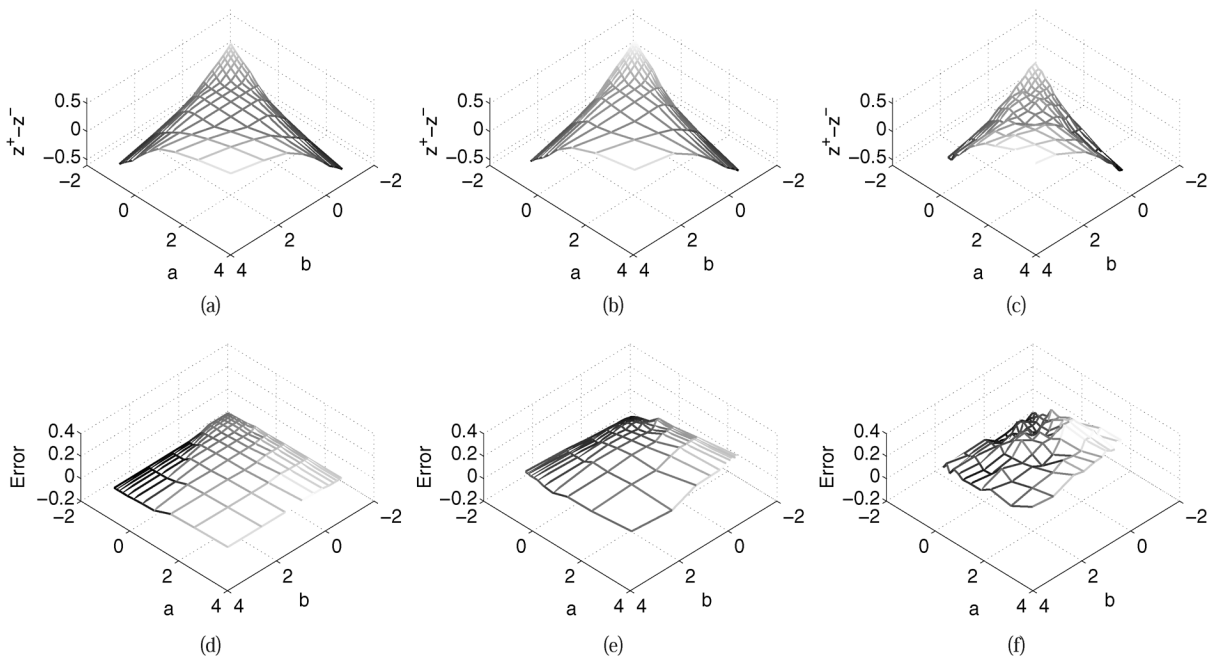


Fig. 14. Multiplication: measurement results for $z^+ - z^-$ with transistors biased in (a) strong inversion; (b) moderate inversion; (c) weak inversion; and (d)–(f) the respective approximation errors.

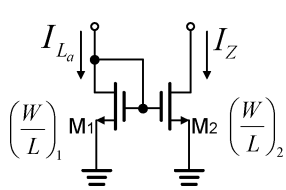


Fig. 15. A current mirror implementing the power function in logarithmic domain.

also claimed in [24] that for MOS transistors the mismatch in V_T dominates the mismatch in β. Therefore, in this paper, our analysis is only be based on the V_T mismatch. The error variance $\sigma^2(\Delta V_T)$ can be converted into an equivalent noise current source in Fig. 17 according to

$$\sigma^2(I) \simeq g_m^2 \sigma^2(\Delta V_T) = g_m^2 \frac{A_{V_T}^2}{WL}. \tag{29}$$

Hence, similar to the noise analysis, the error variance in the output current $\sigma^2(I_z)$ is given by

$$\sigma^2(I_z) \simeq \left(1 + \frac{1}{K} + \frac{1}{g_m^2 R_{\text{source}}^2 K}\right) g_m^2 \frac{A_{V_T}^2}{WL}. \tag{30}$$

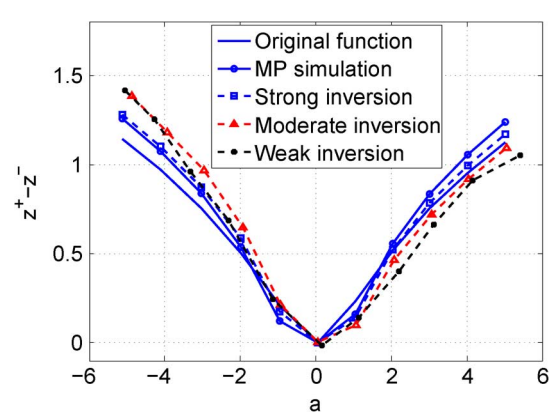


Fig. 16. Power: $z^+ - z^-$ computed according to the log-sum-exp function, the MP approximation simulation and the MP circuit measurement.

Equation (30) shows that for a large K the accuracy of the MP circuit is limited by the mismatch due to the output current mirror stage N_0. Thus, all layout and impedance transformation methods which are used for improving the accuracy of current mirrors, like centroid layout and cascoding techniques,

Authorized licensed use limited to: J.R.D. Tata Memorial Library Indian Institute of Science Bengaluru. Downloaded on December 04,2024 at 10:41:39 UTC from IEEE Xplore. Restrictions apply.

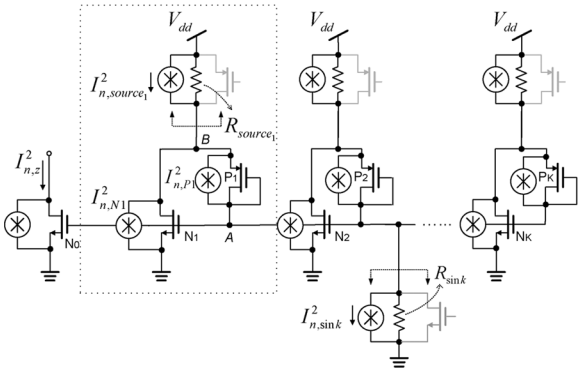


Fig. 17. Noise model for MP circuit in Fig. 4.

can be used to improve the accuracy of MP circuits. The use of cascoding at the output stage transistor N_0 increases the output impedance and makes the core MP module scalable to implement large networks.

B. Speed and Bandwidth Analysis

As with other current-mode circuits, the transient response of the MP circuits is determined by its: a) slew-rate which determines the large-signal response; and b) bandwidth which determines the small-signal response. The node A in Fig. 4 has the largest capacitance and is discharged by the current I_γ. Therefore, the slew-rate of the MP circuit can be expressed as

$$\mathrm{SR}_A = \frac{I_\gamma}{K(C_{\mathrm{GS},N} + C_{\mathrm{GS},P} + C_{\mathrm{DB},P})} \tag{31}$$

where $C_{\mathrm{GS},N}$ represents the gate-to-source capacitance of NMOS transistor N_1 through N_K, and $C_{\mathrm{GS},P}/C_{\mathrm{DB},P}$ denotes the gate-to-source/drain-to-body capacitance of the PMOS transistor P_1 through P_K. Equation (31) shows that the slew-rate will reduce when the number of input branches K increase. However, the small-signal bandwidth of the circuit remains invariant with respect K and is determined by the frequency of the pole located at node A and is given by

$$f_{-3\ \mathrm{dB},A} = \frac{g_{m,P}}{2\pi(C_{\mathrm{GS},N} + C_{\mathrm{GS},P} + C_{\mathrm{DB},P})}. \tag{32}$$

$g_{m,P}$ represents the small-signal gate referred transconductance for the PMOS transistors $P_1 \sim P_K$. The slew-rate and bandwidth analysis show that scaling of MP circuit is limited by the application specific speed requirements, which is controlled by the current I_γ. However, unlike translinear synthesis this limitation can be overcome in MP circuit by rebiasing the circuit in strong inversion.

C. Performance Comparison With Translinear Synthesis

In this section, we summarize and compare the performance of CMOS circuits designed using MP based synthesis with circuits designed using translinear synthesis. The comparisons made here are qualitative in nature because many of the performance metrics (silicon area and power dissipation) are dependent on the topology and complexity of the translinear (TL) circuit.

- **Accuracy:** MP based synthesis relies on PWL approximations at the algorithmic level. However, the accuracy of its circuit level implementation using CMOS current-mode circuit is only limited by mismatch. Our analysis in the previous section showed that the techniques used for designing precision current mirrors can also be used to improve the accuracy of the MP circuits. TL based synthesis, on the other hand, are precise at the algorithmic level. However, their mapping to CMOS current-mode circuits is approximate due to the effect of subthreshold slope (unlike bipolar transistors) and finite drain impedance. These artifacts also introduce temperature dependency in TL circuits, whereas MP circuits are theoretically temperature invariant. Also, due to their operation in weak inversion, CMOS TL circuits are more prone to errors due to mismatch, whereas the MP circuits can be rebiased in strong inversion if higher precision is desired, but at the expense of higher power dissipation.
- **Dynamic range:** The bias-scalable property of the MP circuit enables it to achieve a larger dynamic range compared to a TL circuit. Also, MP synthesis uses currents to represent log-likelihoods, which can also achieve a larger dynamic range.
- **Speed:** For an MP circuit higher speed can be achieved by rebiasing the circuit in strong inversion, without changing the circuit topology. For CMOS translinear circuits higher speed can be achieved by the use of large currents, which implies using large size transistors to maintain their operation in the weak inversion. This will increase the silicon area when compared to an equivalent MP circuit.

V. Application Example: Pattern Classifier

One of the applications where ultra-low power analog computing is attractive is pattern classification [2], [25]. Most pattern classifiers do not require high precision and any analog artifacts (like mismatch and non-linearity) can be calibrated using an offline training procedure. In this example, we use the MP circuits to design a bias-scalable linear classifier. A linear classifier implements an inner-product computation between a weight vector $\mathbf{w} \in \mathbf{R}^n$ and the classifier input $\mathbf{x} \in \mathbf{R}^n$ according to

$$f = \mathbf{w}^T\mathbf{x}. \tag{33}$$

The weight vector $\mathbf{w}$ is determined by an offline training procedure [2] using a labeled training dataset. The two vectors $\overrightarrow{\mathbf{w}}$ and $\overrightarrow{\mathbf{x}}$ are expressed in differential form as

$$\begin{aligned}\overrightarrow{\mathbf{w}} &= \overrightarrow{\mathbf{w}}^+ - \overrightarrow{\mathbf{w}}^- \\ &= [w_1^+, w_2^+, \ldots, w_N^+]^T - [w_1^-, w_2^-, \ldots, w_N^-]^T \\ \overrightarrow{\mathbf{x}} &= \overrightarrow{\mathbf{x}}^+ - \overrightarrow{\mathbf{x}}^- \\ &= [x_1^+, x_2^+, \ldots, x_N^+]^T - [x_1^-, x_2^-, \ldots, x_N^-]^T\end{aligned} \tag{34}$$

with all the $w_i^+(w_i^-)$ and $x_i^+(x_i^-) \in \mathbb{R}^+$. Then (33) can be rewritten as

$$\begin{aligned}&\overrightarrow{\mathbf{w}}^T \cdot \overrightarrow{\mathbf{x}} \\ &= \sum_{i=1}^{N}(w_i^+x_i^+ + w_i^-x_i^-) - \sum_{i=1}^{N}(w_i^+x_i^- + w_i^-x_i^+) \\ &= e^{\log\sum_{i=1}^{N}(w_i^+x_i^+ + w_i^-x_i^-)} - e^{\log\sum_{i=1}^{N}(w_i^+x_i^- + w_i^-x_i^+)} \\ &= e^{z^+} - e^{z^-}\end{aligned} \tag{35}$$

Authorized licensed use limited to: J.R.D. Tata Memorial Library Indian Institute of Science Bengaluru. Downloaded on December 04,2024 at 10:41:39 UTC from IEEE Xplore. Restrictions apply.

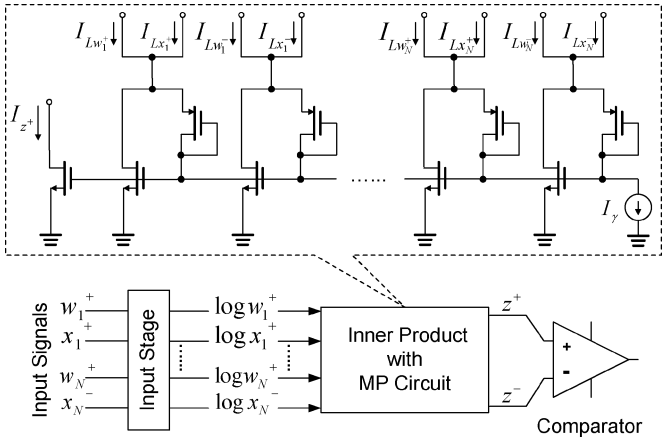


Fig. 18. System architecture and circuit diagram of the MP-based pattern classifier.

when z^+ and z^- are expressed as

$$\begin{aligned} z^+ &= \log \sum_{i=1}^{N} (w_i^+ x_i^+ + w_i^- x_i^-) \\ &\approx M(L_{w_1}{}^+ + L_{x_1}{}^+, L_{w_1}{}^- + L_{x_1}{}^-, \ldots, \\ &\quad L_{w_N}{}^+ + L_{x_N}{}^+, L_{w_N}{}^- + L_{x_N}{}^-, \gamma) \\ z^- &= \log \sum_{i=1}^{N} (w_i^+ x_i^- + w_i^- x_i^+) \\ &\approx M(L_{w_1}{}^+ + L_{x_1}{}^-, L_{w_1}{}^- + L_{x_1}{}^+, \ldots, \\ &\quad L_{w_N}{}^+ + L_{x_N}{}^-, L_{w_N}{}^- + L_{x_N}{}^+, \gamma). \end{aligned} \tag{36}$$

$L_{w_i}{}^+(L_{w_i}{}^-)$ and $L_{x_i}{}^+(L_{x_i}{}^-)$ denote the logarithms of $w_i^+(w_i^-)$ and $x_i^+(x_i^-)$ in (35). For a binary classifier, only the sign of $z^+ - z^-$ is important, implying that the output stage in the architecture (Fig. 1) could be implemented using a comparator (instead of $e^{z^+} - e^{z^-}$). The system architecture and circuit diagram for the pattern classifier is shown in Fig. 18.

In this example, we generated a synthetic linearly separable dataset corresponding to classes: class I and class II. We used a linear support vector machine training algorithm [13] to determine weight $\mathbf{w}$. The parameter vector $\mathbf{w}$ is then programmed as currents on the prototype chip. Test vectors $\mathbf{x}$ were selected randomly and were then applied as input to the chip. For the sake of visualization, we show only results from a two-dimensional dataset. In this special case, the classifier equation (33) reduces to a two-dimensional inner product formulation as described in the example in Section I. The circuit implementation is therefore identical to that of a full quadrant multiplier in Fig. 12. Fig. 19 shows the classification boundary and the classifier scores (negative values are represented by darker shades, whereas positive values are represented by lighter shades) obtained from software simulations and from the fabricated prototype under different biasing conditions. Again, the results show that the operation of the prototype is bias-scalable and the classification performance has been verified to be similar to that of the software model.

VI. Conclusions and Discussions

Unlike translinear circuits, current-mode circuits which rely only on universal conservation principles like the KCL are portable across different biasing regions of a MOS transistor. In this paper, we proposed an MP based synthesis technique which can be used to design analog circuits that are bias scalable. At the core of MP synthesis is a PWL approximation of the log-sump-exp function and in this paper we have shown that many mathematical functions can be derived in terms of log-sum-exp and hence MP functions. MP circuits use only addition, subtraction, and threshold operations to approximate log-sum-exp functions and hence can be implemented using a few transistors. The main advantages of the proposed MP synthesis technique are summarized as follows:

- MP circuits can operate over a wider dynamic range as the operation of the transistor can span different biasing regions.
- MP synthesis provide a systematic approach for designing MOS circuits operating in the moderate inversion region.
- The use of KCL and thresholding ensures that MP circuits are insensitive to temperature variations.
- MP circuits operate in the log-likelihood domain where complex functions such as multiplication, division, square-root operations are mapped into simpler operations like addition, subtraction, or scaling, making circuit design less complex. Thus, large-scale analog systems which use non-linear functions (e.g., support vector machines [2]) can be easily implemented using MP circuits.
- The operation of the MP circuits in the log-likelihood domain enhances the dynamic range of the circuit which is important for analog recursive computations like the ones used in hidden Markov model (HMM) decoding.

However, there are several open challenges which we envision will be a part of the future work in this area. The role of the hyper-parameter γ in the circuit synthesis has not been fully explored. It is anticipated that γ will play a key role in controlling the trade-off between the approximation error and the power dissipation of the MP circuit. Also, the application of MP-based synthesis can be extended to other types of analog circuits like charge-mode or time-domain circuits by using other universal conservation principles like charge conservation. Like translinear synthesis, MP synthesis could potentially be extended towards designing dynamic analog circuits like filters. Also, extending this work to operate with real and imaginary quantities would also form a part of the future work.

Appendix A
Proof: MP as PWL Approximation of the Log-Sum-Exp Function

Proof: Given a set of operands $L_1, \ldots, L_N$, $L_i \in \mathbb{R}, i = 1, \ldots, N$, the generalized form of log-sum-exp function is given by

$$z_{\log} = \log \left(\sum_{i=1}^{N} e^{L_i} \right). \tag{37}$$

Authorized licensed use limited to: J.R.D. Tata Memorial Library Indian Institute of Science Bengaluru. Downloaded on December 04,2024 at 10:41:39 UTC from IEEE Xplore. Restrictions apply

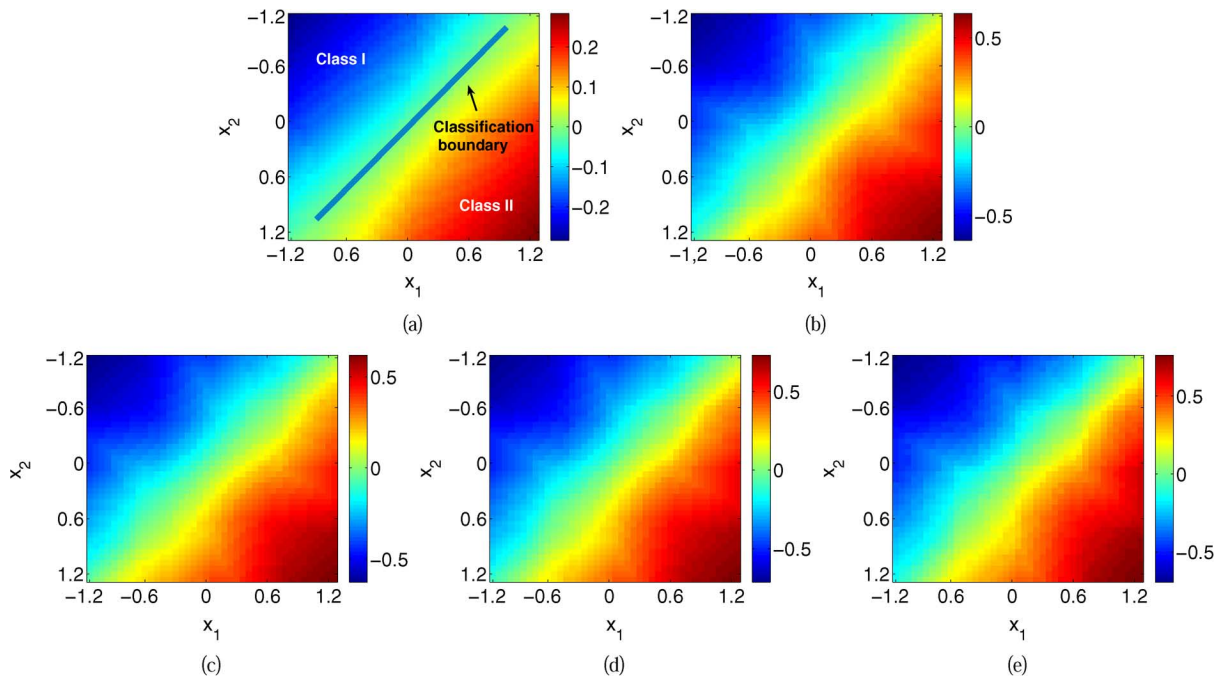


Fig. 19. Classifier: $z^+ - z^-$ simulated by: (a) log-sum-exp function (simulation); (b) MP approximation (simulation); and measured $z^+ - z^-$ when the transistors are biased in (c) strong inversion (measurement); (d) moderate inversion (measurement); (e) weak inversion (measurement).

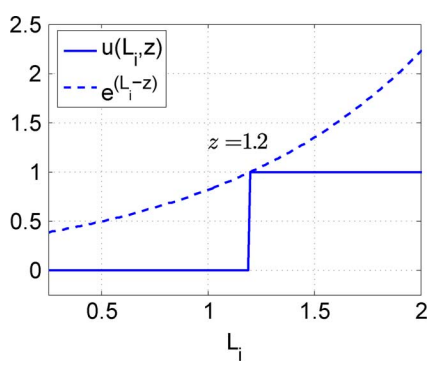


Fig. 20. The curve of $e^{(L_i - z)}$ and its PWL approximation curve of $u(L_i, z)$.

The derivative of $z_{\log}$ with respect to one of the operands $L_k, 1 \le k \le N$ is given by

$$\frac{\partial z_{\log}}{\partial L_k} = \frac{e^{L_k}}{\sum_{i=1}^{N} e^{L_i}} = \frac{e^{L_k - z_{\log}}}{\sum_{i=1}^{N} (e^{L_i - z_{\log}})}. \tag{38}$$

We introduce a variable z and define a step function $u(L_i, z)$ according to

$$u(L_i, z) = \begin{cases} 1, & L_i \ge z, \\ 0, & L_i < z, \end{cases} \quad 1 \le i \le N.$$

It can be readily verified (shown in Fig. 20) that $u(L_i, z_{\log})$ is a lower-bound for the function $e^{(L_i - z_{\log})}$. Also, the use of step-functions to approximate gradients will lead to a piecewise linear functions when integrated. Equation (38) can be approximated as

$$\frac{\partial z_{\log}}{\partial L_k} \approx \frac{\partial z}{\partial L_k} = \frac{u(L_k, z)}{\sum_{i=1}^{N} u(L_i, z)}. \tag{39}$$

Because the differential equation (39) consists of discontinuities, we will find its solution only in the continuous regions. For instance, we will assume that the integral is computed in the region where $\sum_{i=1}^{N} u(L_i, z)$ which is constant. The solution to (39) can be written as

$$z = \frac{\sum_i [L_i u(L_i, z)] - \gamma}{\sum_{i=1}^{N} u(L_i, z)} \tag{40}$$

where γ is an constant of integration. Equation (40) leads to

$$\sum_{i=1}^{N} [L_i - z] u(L_i, z) = \gamma \tag{41}$$

$$\sum_{i=1}^{N} [L_i - z]_+ = \gamma \tag{42}$$

which is the MP function $z = M(L_1, \ldots, L_N, \gamma)$. ■

Appendix B
MP Synthesis of Power Functions

For an operand a, $a \in \mathbb{R}$, its power function is expressed as a^n, $n \in \mathbb{Z}^+$. As illustrated in Section III, a is expressed in its differential form $a = a^+ - a^-, a^+(a^-) \in \mathbb{R}^+$ and $a^+(a^-) > 1$. For arbitrary n, a^n can be written as

$$\begin{aligned} &(a^+ - a^-)^n \\ &\quad = \sum_i (-1)^i C_n^i (a^+)^{n-i} (a^-)^i \\ &\quad = \sum_{i:\text{even}} C_n^i (a^+)^{n-i} (a^-)^i - \sum_{i:\text{odd}} C_n^i (a^+)^{n-i} (a^-)^i \\ &\quad = e^{\log\left(\sum_{i:\text{even}} C_n^i (a^+)^{n-i} (a^-)^i\right)} \\ &\qquad - e^{\log\left(\sum_{i:\text{odd}} C_n^i (a^+)^{n-i} (a^-)^i\right)}. \end{aligned} \tag{43}$$

Authorized licensed use limited to: J.R.D. Tata Memorial Library Indian Institute of Science Bengaluru. Downloaded on December 04,2024 at 10:41:39 UTC from IEEE Xplore. Restrictions apply.

Designate z^+ as $\log(\sum_{i:\text{even}} C_n^i (a^+)^{n-i}(a^-)^i)$ and z^- to represent $\log(\sum_{i:\text{odd}} C_n^i (a^+)^{n-i}(a^-)^i)$ in (43). Then

$$\begin{aligned} z^+ &\approx M(\mathcal{L}^+, \gamma). \\ z^- &\approx M(\mathcal{L}^-, \gamma). \end{aligned} \tag{44}$$

$\mathcal{L}^+(\mathcal{L}^-)$ is a group of L_is satisfying

$$L_i = \log C_n^i + (n-i)\log a^+ + i \log a^-, \tag{45}$$

$i \in [0, n]$ and i is even (odd).

REFERENCES

[1] E. A. Vittoz, "Low power design: Ways to approach the limits," in *IEEE ISSCC 1994 Dig. Tech. Papers*, San Francisco, CA, pp. 14–18.

[2] S. Chakrabartty and G. Cauwenberghs, "Sub-microwatt analog VLSI trainable pattern classifier," *IEEE J. Solid-State Circuits*, vol. 42, no. 5, pp. 1169–1179, May 2007.

[3] G. E. R. Cowan, R. C. Melville, and Y. P. Tsividis, "A VLSI analog computer/digital computer accelerator," *IEEE J. Solid-State Circuits*, vol. 41, no. 1, pp. 42–53, Jan. 2006.

[4] B. Gilbert, "Translinear circuits: A proposed classification," *Electron. Lett.*, vol. 11, no. 1, pp. 14–16, 1975.

[5] E. Seevinck, R. F. Wassenaar, and H. C. K. Wong, "A wide-band technique for vector summation and rms-dc conversion," *IEEE J. Solid-State Circuits*, vol. SSC-19, no. 3, pp. 311–318, 1984.

[6] A. G. Andreou and K. A. Boahen, "Translinear circuits in subthreshold MOS," *Analog Integr. Circuits Signal Process.*, vol. 9, no. 2, pp. 141–166, 1996.

[7] T. Serrano-Gotarredona, B. Linares-Barranco, and A. G. Andreou, "A general translinear principle for subthreshold MOS transistors," *IEEE Trans. Circuits Syst. I, Fundam. Theory Appl.*, vol. 46, no. 5, pp. 607–616, May 1999.

[8] B. A. Minch, "Synthesis of static and dynamic multiple-input translinear element networks," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 51, no. 2, pp. 409–421, Feb. 2004.

[9] E. Seevinck and R. J. Wiegerink, "Generalized translinear circuit principle," *IEEE J. Solid-State Circuits*, vol. 26, no. 8, pp. 1098–1102, 1991.

[10] B. A. Minch, "MOS translinear principle for all inversion levels," *IEEE Trans. Circuits Syst. II, Exp. Briefs*, vol. 55, no. 2, pp. 121–125, Feb. 2008.

[11] C. C. Enz, F. Krummernacher, and E. A. Vittoz, "An analytical MOS transistor model valid in all regions of operation and dedicated to low-voltage and low-current applications," *Analog Integrat. Circuits Signal Process. J.*, vol. 8, pp. 83–114, Jul. 1995.

[12] F. Silveira, D. Flandre, and P. G. A. Jespers, "A floating-gate MOS learning array with locally computed weight updates," *IEEE J. Solid-State Circuits*, vol. 31, no. 9, pp. 1314–1319, Sep. 1996.

[13] S. Chakrabartty and G. Cauwenberghs, "Gini-support vector machine: Quadratic entropy based multi-class probability regression," *J. Mach. Learn. Res.*, vol. 8, pp. 813–839, 2007.

[14] M. Gu, K. Misra, H. Radha, and S. Chakrabartty, "Sparse decoding of low density parity check codes using margin propagation," in *Proc. IEEE Globecom*, Nov. 2009.

[15] K. L. Hsiung, S. J. Kim, and S. Boyd, "Tractable approximate robust geometric programming," *Optim. Eng.*, vol. 9, no. 2, pp. 95–118, June 2008.

[16] T. M. Cover and J. A. Thomas, *Elements of Information Theory*, 2nd ed. New York: Wiley-IEEE, 2006.

[17] H.-A. Loeliger, F. Lustenberger, M. Helfenstein, and F. Tarköy, "Probability propagation and decoding in analog VLSI," *IEEE Trans. Inf. Theory*, vol. 47, no. 2, pp. 837–843, Feb. 2001.

[18] C. Winstead, N. Nguyen, V. Gaudet, and C. Schlegel, "Low-voltage CMOS circuits for analog iterative decoders," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 53, pp. 829–841, Apr. 2006.

[19] M. Gu and S. Chakrabartty, "A 100 pJ/bit, (32,8) CMOS analog low-density parity-check decoder based on margin propagation," *IEEE J. Solid-State Circuits*, vol. 46, no. 6, pp. 1433–1442, Jun. 2011.

[20] S. Hemati, A. Banihashemi, and C. Plett, "An 80-Mb/s 0.18-μm CMOS analog min-sum iterative decoder for a (32,8,10) LDPC code," *IEEE J. Solid-State Circuits*, vol. 41, pp. 2531–2540, Nov. 2006.

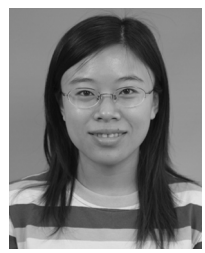
[21] Y.-H. Lam, W.-H. Ki, and C.-Y. Tsui, "Integrated low-loss CMOS active rectifier for wirelessly powered devices," *IEEE Trans. Circuits Syst. II, Exp. Briefs*, vol. 53, no. 12, pp. 1378–1382, Dec. 2006.

[22] A. Gore, S. Chakrabartty, S. Pal, and E. C. Alocilja, "A multichannel femtoampere-sensitivity potentiostat array for biosensing applications," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 53, no. 11, pp. 2357–2363, Nov. 2006.

[23] B. Razavi, *Design of Analog CMOS Integrated Circuits*. New York: McGraw-Hill, Aug. 2000.

[24] P. R. Kinget, "Device mismatch and tradeoffs in the design of analog circuits," *IEEE J. Solid-State Circuits*, vol. 40, no. 6, pp. 1212–1224, Jun. 2005.

[25] K. Kang and T. Shibata, "An on-chip-trainable gaussian-kernel analog support vector machine," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 57, no. 7, pp. 1513–1524, Jul. 2010.



Ming Gu received the B.E. and M.E. degrees from the School of Automation Science and Electrical Engineering, Beijing University of Aeronautics and Astronautics, Beijing, China, in 2000 and 2003, respectively, and the M.E. degree in electrical and computer engineering from Michigan State University, East Lansing, in 2008. She is currently working toward the Ph.D. degree at Michigan State University.

From 2003 to 2006, she worked as a Research Engineer in the Center for Space Science and Applied Research, Chinese Academy of Sciences, Beijing.

Her research interests include mixed signal VLSI circuits design, error-control coding, and analog signal processing.

Shantanu Chakrabartty (S'99–M'04–SM'09) received the B.Tech. degree from Indian Institute of Technology, Delhi, in 1996 and the M.S. and Ph.D. degrees in electrical engineering from Johns Hopkins University, Baltimore, MD, in 2002 and 2005 respectively.

From 1996 to 1999 he was with Qualcomm Incorporated, San Diego, CA, and during 2002 he was a Visiting Researcher at the University of Tokyo, Japan. He is currently an Associate Professor in the Department of Electrical and Computer Engineering at Michigan State University (MSU), East Lansing. His work covers different aspects of analog computing, in particular nonvolatile circuits, and his current research interests include energy harvesting sensors and neuromorphic and hybrid circuits and systems.

Dr. Chakrabartty was a Catalyst foundation fellow from 1999 to 2004 and is a recipient of National Science Foundations CAREER award and University Teacher-Scholar Award from MSU. He is currently serving as the Associate Editor for IEEE TRANSACTIONS OF BIOMEDICAL CIRCUITS AND SYSTEMS, Associate Editor for the *Advances in Artificial Neural Systems*, and a Review Editor for *Frontiers of Neuromorphic Engineering*.

Authorized licensed use limited to: J.R.D. Tata Memorial Library Indian Institute of Science Bengaluru. Downloaded on December 04,2024 at 10:41:39 UTC from IEEE Xplore. Restrictions apply