

A Bit-Parallel Deterministic Stochastic Multiplier

Sairam Sri Vatsavai and Ishan Thakkar

Department of Electrical and Computer Engineering, University of Kentucky, Lexington, KY 40506, USA

Abstract—This paper presents a novel bit-parallel deterministic stochastic multiplier, which improves the area-energy-latency product by up to 10.6×10^4 , while improving the computational error by 32.2%, compared to three prior stochastic multipliers.

I. INTRODUCTION

Stochastic Computing (SC) is an unconventional form of computing where numbers are represented by the probability of observing a '1' in bit-streams called stochastic bit-streams (SBs) [1]. In SC's unipolar format, W is an SB of N bits that represents a real-valued variable $v \in [0, 1]$, $v = N_1/N$, where N_1 is the number of '1's in W . SC offers a low-cost multiplication using a standard logic AND gate [1]. Therefore, stochastic multipliers can decrease the hardware complexity of GEMM circuits used in deep learning accelerators [2]. **But** Stochastic multipliers suffer from computational errors. To reduce errors, prior works [1], [3] used lengthy pseudo-random SBs. In contrast, [2] showed that errors can be minimized by deterministically re-adjusting the bit-position correlations in randomly generated SBs. Atop errors, prior stochastic multipliers also suffer from very high latency and energy due to their use of lengthy SBs and bit-serial operation. To address both of these challenges, we present, *for the first time*, a novel stochastic multiplier that generates SBs with reduced lengths and deterministic bit-position correlations in a bit-parallel manner, thereby simultaneously minimizing the latency, energy, and errors.

II. OUR STOCHASTIC MULTIPLIER

Fig. 1(a) shows our stochastic multiplier, which first converts two B-bit binary operands (X_b and Y_b) to N-bit SBs ($X_u = [x_u^N, \dots, x_u^1]$ and $Y_u = [y_u^N, \dots, y_u^1]$), where $N=2^B$. Subsequently, it performs bit-wise AND on X_u and Y_u to obtain the stochastic multiplication result $O_u = [o_u^N, \dots, o_u^1]$. To minimize the errors, the conditional probability $P(Y_u|X_u)$ must be equal to the marginal probability $P(X_u)$ [2]. Our multiplier achieves that as follows. From Fig. 1(a), for operand X_b , a binary-to-transition-coded-unary (B-to-TCU) decoder generates all N bits of X_u with '1's grouped at the trailing end (on the right). For operand Y_b , only the binary bits $[y_b^{B-1}, \dots, y_b^1]$ of Y_b go to the B-to-TCU decoder to generate bits $[y_u^{B-1}, \dots, y_u^1]$. These bits together with y_b^B propagate through the bit-position correlation encoder (the array of AND and OR gates) to generate Y_u , while maintaining $P(Y_u|X_u) = P(X_u)$. Once X_u and Y_u are available, they are pushed through the array of AND gates to obtain O_u . Table I reports examples of how our proposed design generates X_u and Y_u , and then multiplies them using AND gates to generate O_u .

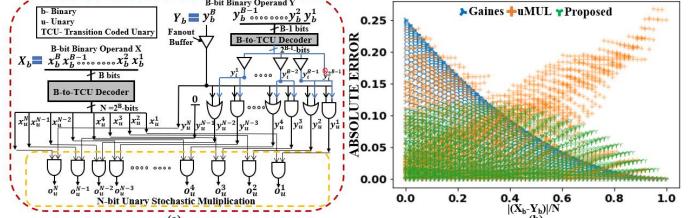


Fig. 1: (a) Schematic of our proposed stochastic multiplier, (b) distribution of absolute error in various stochastic multipliers.

TABLE I: EXAMPLES FOR OUR STOCHASTIC MULTIPLIER. ERROR IS THE DIFFERENCE BETWEEN THE TARGET AND ACTUAL OUTPUT PROBABILITIES.

$X_u = P(x_u=1)$	$Y_u = P(y_u=1)$	$O_u = P(o_u=1)$	Error
00001111=4/8	10111110=6/8	00001110=3/8	0
00011111=5/8	00101010=3/8	000001010=2/8	0.01
00000111=3/8	10101010=4/8	00000010=1/8	0.06

TABLE II: COMPARISON OF STOCHASTIC MULTIPLIERS. A=AREA, L=LATENCY, E=ENERGY, MAE=MEAN ABSOLUTE ERROR

Unit	A (μm^2)	L (ns)	E \times L (pJ.s)	A \times E \times L (pJ.s.mm 2)	MAE
uMUL [2]	207.6	640	2.5E-08	5.2E-09	0.06
Gaines [1]	378.7	640	4.9E-08	1.9E-08	0.08
Jenson [3]	520.2	163840	3.5E-03	1.8E-03	0.07
Proposed	540.6	0.17	9.2E-14	4.9E-14	0.04

III. EVALUATION

Table II reports the hardware costs and Mean Absolute Error (MAE) for various multipliers for $B=8$ -bit. Our proposed multiplier achieves 32.2%, 42.8%, and 51.8% lower MAE compared to uMUL [2], Jenson [3], and Gaines [1], respectively. In addition, our proposed multiplier achieves 10.6×10^4 better area-energy-latency product compared to the best prior work uMUL [2]. Moreover, we also show in Fig. 1(b) that the absolute error in the multiplication results from our multiplier is less dependent on the normalized difference of input operands ($(|X_b - Y_b|)/N$). This implies that our multiplier can provide stable accuracy irrespective of the input operand values, which is a desirable quality to have in multipliers used in GEMM accelerators.

IV. CONCLUSIONS

We presented a novel stochastic multiplier that generates stochastic bit-streams with reduced lengths and deterministic bit-position correlations in a bit-parallel manner, thereby simultaneously minimizing the latency, energy, and errors.

REFERENCES

- [1] B. R. Gaines, *Stochastic Computing Systems*, 1969.
- [2] D. Wu *et al.*, “Ugemm: Unary computing architecture for gemm applications,” in *ISCA*, 2020.
- [3] D. Jenson *et al.*, “A deterministic approach to stochastic computation,” in *ICCAD*, 2016.