FeReX: A Reconfigurable Design of Multi-bit Ferroelectric Compute-in-Memory for Nearest Neighbor Search

Zhicheng Xu[1,2], Che-Kai Liu[3], Chao Li[2], Ruibin Mao[1], Jianyi Yang[2,4,*],
Thomas Kämpfe[5], Mohsen Imani[6], Can Li[1*], Cheng Zhuo[2,7,*] and Xunzhao Yin[2,7,*]
[1]Department of Electrical and Electronic Engineering, The University of Hong Kong
[2]Zhejiang University; [3]School of Electrical and Computer Engineering, Georgia Institute of Technology
[4]ZJU-Hangzhou Global Scientific and Technological Innovation Center; [5]Center Nanoelectric Technologies, Fraunhofer IPMS
[6]Department of Computer Science, University of California Irvine; [7]Key Lab of CS&AUS of Zhejiang Province
[*]Corresponding authors, email: {czhuo, yangjy, xzyin1}@zju.edu.cn; canl@hku.hk

***Abstract*—Rapid advancements in artificial intelligence have given rise to transformative models, profoundly impacting our lives. These models demand massive volumes of data to operate effectively, exacerbating the data-transfer bottleneck inherent in the conventional von-Neumann architecture. Compute-in-memory (CIM), a novel computing paradigm, tackles these issues by seamlessly embedding in-memory search functions, thereby obviating the need for data transfers. However, existing non-volatile memory (NVM)-based accelerators are application specific. During the similarity based associative search operation, they only support a single, specific distance metric, such as Hamming, Manhattan, or Euclidean distance in measuring the query against the stored data, calling for reconfigurable in-memory solutions adaptable to various applications. To overcome such a limitation, in this paper, we present FeReX, a reconfigurable associative memory (AM) that accommodates various distance metrics including Hamming, Manhattan, and Euclidean distances. Leveraging multi-bit ferroelectric field-effect transistors (FeFETs) as the proxy and a hardware-software co-design approach, we introduce a constrained satisfaction problem (CSP)-based method to automate AM search input voltage and stored voltage configurations for different distance based search functions. Device-circuit co-simulations first validate the effectiveness of the proposed FeReX methodology for reconfigurable search distance functions. Then, we benchmark FeReX in the context of k-nearest neighbor (KNN) and hyperdimensional computing (HDC), which highlights the robustness of FeReX and demonstrates up to 250× speedup and 10^4 energy savings compared with GPU.**

I. Introduction

The artificial intelligence (AI) models yield a profound influence over various aspects of our lives. These models, however, frequently require vast amounts of data for their operation, thus exacerbating the data-transfer bottleneck inherent in the traditional von Neumann architecture. Consequently, there is a growing demand for a departure from the conventional computing paradigm, one that seamlessly integrates the critical functionalities of emerging AI models within the memory itself. This shift is not only desirable but also essential to keep pace with the demands of modern computing.

Compute-in-memory (CIM) has emerged as an alternative computing paradigm that integrates the separated computing unit and memory that exists in Von Neuman machine altogether [1]–[5]. Several CIM primitives, i.e., associative memories (AMs) that support various distance metric computations between input and stored vectors have demonstrated their potential for accelerating similarity based inferences in novel machine learning algorithms [6]–[12]. Hamming distance (HD)-based CIM design has been originally proposed [13] for memory-augmented neural networks (MANN), but it suffers from non-negligible classification accuracy degradation. Recently, CIM design that implements Manhattan distance for MANN classification has been experimentally verified [14], and CIM design realizing Euclidean distance for hyperdimensional computing (HDC) has been demonstrated at the device level [15]. These CIM based AM designs aim to address the non-negligible algorithmic accuracy degradation with complex distance functions used in a certain application. However, existing non-volatile memory (NVM)-based AMs are limited to a specific classification task, as one AM design can only support a single distance computation, such as Hamming [13], [16], [17], Manhattan [14], Euclidean [15], and sigmoid [18]. A CIM search engine that can achieve a reconfigurable distance function is highly desirable. Based on the nature of various applications, different distance functions may be used during the similarity based search, and, within a certain application, several distance functions may be exploited for various datasets.

In this paper, we propose FeReX, a reconfigurable CIM-based AM for Hamming, Manhattan, and Euclidean distance searches, utilizing multi-bit ferroelectric field-effect transistor (FeFET) devices as the proxy. We propose a hardware-software co-design scheme to efficiently realize similarity searches between a query and stored vectors in terms of various distance metrics. This involves constructing a matrix of target distance values between the query and stored vectors based on the given distance function. To accommodate this target matrix, we formulate a constrained satisfaction problem (CSP), which incorporates the FeFET device and crossbar constraints related to the output currents, input voltages and stored threshold voltages. By solving the CSP using backtracking and AC-3 algorithm, we find the optimal search input and stored voltage configurations for the input query and stored vectors that align the CSP formulation with the target distance matrix. In this sense, FeReX can be readily configured to support a range of distance functions in an automated way. FeReX incorporates a Loser-Take-All (LTA) circuit structure, enabling it to support nearest neighbor search functionality. Our extensive

979-8-3503-4859-0/DATE24/© 2024 EDAA

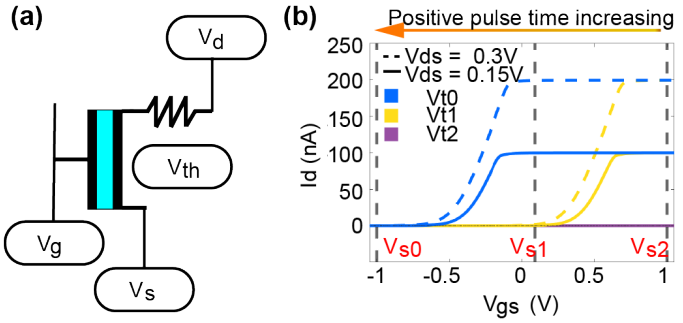

Fig. 1. (a) 1FeFET1R structure. (b) multi-level I-V curve of 1FeFET1R, where V_{t0}, V_{t1}, V_{t2} represent different V_{th} stored in the FeFET, V_{s0}, V_{s1}, V_{s2} represent different search voltage (i.e, V_{gs}) applied to the FeFET, and two different V_{ds} result in two level of ON currents.

performance assessment in the realms of k-nearest neighbor (KNN) and HDC applications underscores the robustness and efficacy of our design approach, highlighting its resilience and efficiency. Notably, FeReX achieves up to 250× speedup and 10^4 energy savings compared to GPU implementations. To best of our knowledge, this work represents the first reconfigurable distance search implementation within NVM-based AM.

II. Background

In this Sec., we review the FeFET characteristics that are exploited within the FeReX. Then, recent AM designs for NN search are briefly summarized.

A. FeFET Characteristics

Excellent CMOS compatibility, outstanding scalability, and superior energy efficiency [19] of HfO_2 ferroelectric materials elucidate the competitiveness of Ferroelectric FET (FeFET) among other NVMs. Based on the conventional CMOS transistor, a FeFET is made with ferroelectric materials integrated into the CMOS gate stack. The stored value is represented by the threshold voltage (V_{th}) of a FeFET, and can be altered by applying a positive or negative voltage pulse at the device gate, which in turn changes the polarization of the Fe layer. Specifically, the value of V_{th} is determined by the duration and magnitude of the applied voltage pulse [13]. For instance, if the duration of a given positive voltage pulse increases, the V_{th} will shift lower accordingly.

Recently, Soliman et al. propose a cell that integrates a resistor with a single FeFET [20], as shown in Fig 1(a). It is demonstrated in [20], [21] that by connecting a large resistor at the source (or equivalently, drain) of the FeFET, the ON state current I_{ds} is significantly reduced and thus is independent of V_{th} variation [20]. Saito et al. further demonstrate a back-end-of-line (BEOL) 1FeFET1R structure, incurring no additional area penalty with an $M\Omega$ resistor integrated with a FeFET [22]. Given a V_{ds} and resistance R, The conducting current of a FeFET can be approximated as Min$\{I_{sat}, V_{ds}/R\}$ due to the fact that it is possible when $I_{ds} = V_{ds}/R$ under a given V_{gs}, the FeFET operates in the linear region. In this work, all V_{ds} values are integer multiples of the minimum V_{ds} value, ensuring that all I_{ds} values are interger multiples of the minimum I_{ds} value. Fig. 1(b) illustrates a multi-level cell (MLC) 1FeFET1R characteristics. When $V_{gs} > V_{th}$, various V_{th} and V_{gs} values can be explored, where I_{ds} is approximately equivalent to V_{ds}/R, while I_{ds} approaches 0 if $V_{gs} < V_{th}$.

TABLE I
Existing AMs with Different Distance Functions

Design	NVM	Cell structure	MLC	Distance function
Nat. Ele. [23]	PCM	1PCM	No	Hamming
IEDM'20 [24]	FeFET	2FeFET-1T	Yes	Best-match
TED'21 [14]	RRAM	2RRAM	Yes	Manhattan
TC'21 [18]	FeFET	2FeFET	Yes	Sigmoid
SR'22 [15]	FeFET	2FeFET	Yes	Euclidean
FeReX (This work)	FeFET	1FeFET-1R	Yes	HD/L_1/L_2

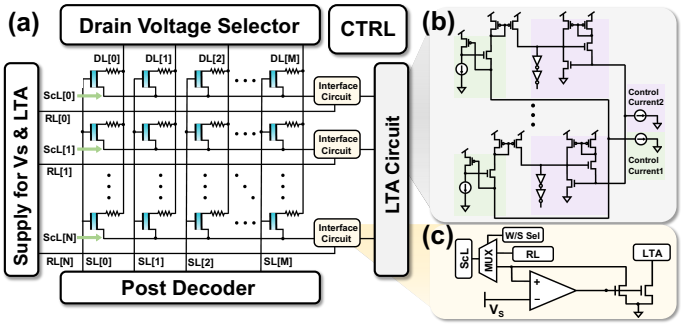

Fig. 2. (a) FeReX AM overview. (b) LTA and (c) Interface circuit.

B. Existing AM Designs

AM has been deployed in a variety of scenarios such as HDC [23], [25], [26], MANN [14], few-shot learning [18], and so on. Table I summarizes existing AMs based on single-level cell/multi-level cell (SLC/MLC) NVMs with different distance functions. A matching-based MLC 2FeFET-1T AM has been fabricated in [24]. To further achieve algorithmic level accuracy, AMs with intricate distance functions utilizing MLC cells have been proposed including sigmoid and Euclidean functions [15], [18], etc. However, these efforts are typically designed for a fixed distance function. In this work, FeReX is able to support multiple distance functions as shown in Tab. I. Below we elaborate on the designed AM and its peripherals first. Then, the proposed encoding scheme for selecting the search input voltages and programming the V_{th} voltages is elucidated in Sec. III-B.

III. FeReX: Reconfigurable In-Memory Search Engine

A. FeReX Circuit Design

In this subsection, we briefly describe FeReX, the FeFET-based AM design along with its distance sensing peripherals. The peripherals for the array includes the level shifters for high write voltages, column switch matrix for selecting columns and input decoder (or digital-to-analog converter) [27]. Fig. 2 shows the detailed circuit schematic of the proposed FeReX. FeReX consists of a 1FeFET1R based crossbar array with the drain voltage selector and the interface sensing circuit blocks for each row. The loser-take-all (LTA) circuitry compares the currents

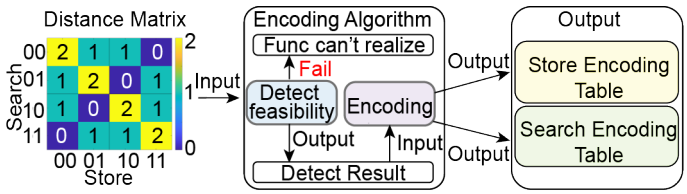

Fig. 3. Workflow of FeReX's encoding scheme.

from array rows to perform nearest neighbor search operation. The search lines (SLs) and drain lines (DLs) are shared by the FeFETs within the same column, and the source lines (ScLs) link the FeFETs within the same row, as shown in Fig. 2(a).

During the write/erase phase, the MUX of interface circuit selects row lines (RLs), and $V_{ScL} = V_{RL}$. In this configuration, the RL voltage of the selected row is 0V, while the RL voltage of the unselected rows is raised to half of V_{write}/V_{erase}. Such writing inhibition scheme prevents write disturbance [28]. During the search phase, search voltages are applied to FeFET gates through SLs, and the MUX in the interface circuit selects the op-amp, setting all voltages on ScLs to V_s. As can be seen from Fig. 1, the FeFET's ON state current flows from the DL to the ScL only when the applied search voltage V_{search} at the gate exceeds the stored threshold voltage V_{th}. Otherwise, the FeFET remains in the cut-off state. The ON current I_{ON} through the FeFET is determined by the voltage drain-source voltage V_{ds}, as discussed in Sec. II. Given that all FeFETs sharing the same DL experience the same V_{ds}, the ON currents I_{ON} through the activated FeFETs within the same column are identical. The currents flowing through FeFETs in the same row are aggregated at ScL and sensed by interface circuit. The op-amps of all rows are used to inhibit ScL voltage fluctuation, as the change in V_{ds} of FeFETs will alter the I_{ON} accordingly, resulting in inaccurate LTA sensing. LTA circuit compares the row currents and indicates the rwo with the minimal current. The operation of current domain LTA circuitry is similar to winner-take-all (WTA), which has been utilized for NN detection as well. Readers interested in detailed explanations can refer to [29].

B. FeReX Encoding Algorithm

Fig. 3 depicts the overview of our proposed encoding algorithm that finds the search and stored voltage configurations for FeReX array given a distance function. The target distance values between the query and stored vectors are first constructed as a function table matrix. Then a CSP incorporating the FeFET constraints is formulated to determine whether the target distance matrix can be achieved using FeReX array. The query and stored voltage configurations are encoded by addressing the CSP to align with the target distance matrix.

Unlike conventional AM designs that consist of a fixed number of NVM devices per cell, FeReX flexibly configures the number of FeFETs in each AM cell to represent the data vectors. The distance metrics can be represented by the Distance Matrix (DM). Within the matrix, columns stand for stored values, and rows correspond to various search values, with each element in the matrix denoting the distance between a stored value and a search value. Fig. 4(a) shows the DM corresponding to a 2-bit Hamming distance, i.e., the distance between the input search vector '00' and store vector '11' is 2.

Fig. 4(b) illustrates the search and stored data encoding to the FeReX circuit. The stored encoding is represented by V_{th} value in each FeFET device, while the search encoding consists of the FeFET's V_{ds} and V_{gs} voltages. V_{ds} determines the current flowing through the FeFET when the FeFET is activated as shown in Fig. 1(b). The total current flowing through a cell of FeReX represents the distance value between the stored value and input value, i.e., the DM element value. In order to implement the DM using the FeReX cell, we need to figure out the search and stored encoding configuration within a cell. Without loss of generality, the number of FeFETs per cell is k, the DM element value at row sch and column sto is denoted as $I_{sch,sto}$, and the current flowing through the FeFET i under search sch and stored sto value condition is $I_{sch,sto,i}$. Implementing the DM based on a FeReX cell involves solving a constrained satisfaction problem (CSP) with three constraints.

Fig. 4(c) illustrates the representation of a DM element $I_{sch,sto}$ '2' by the currents of a set of FeFETs DMCurs[sch, sto] (three FeFETs are used in this example). The implementation decomposes the element $I_{sch,sto}$ into decomposed values, i.e., $I_{sch,sto} = \sum_{i=1}^{k} I_{sch,sto,i}$, where $I_{sch,sto,i}$='0' indicates FeFET i is at OFF-state, and $I_{sch,sto,i}$='1/2' indicates that FeFET i is activated with multi-level V_{ds}s as shown in Fig. 1(b). Since the current of FeFET i under stored sto and search sch condition, i.e., $I_{sch,sto,i}$ represents the value between '0' and the maximal DM value, and the number of possible $I_{sch,sto,i}$ values is limited per FeFET's operating condition, the possible FeFET currents $I_{sch,sto,i}$ representing DM element $I_{sch,sto}$ are constrained, forming the set DMCurs[sch, sto]. We refer to this constraint as the first constraint.

Secondly, considering that the FeFET i under search sch condition should either conduct the identical ON current, or be at OFF state, the current of this FeFET under different sto conditions should be the same or 0, i.e., $I_{sch,sto_a,i} = I_{sch,sto_b,i}$ or 0, $\forall a, b \in sto$. For example, as shown in Fig. 4(d), $I_{Search11,Store00,1}$ must be equal to $I_{Search11,Store01,1}$ or 0. We refer to this as the second constraint.

The third constraint arises from the multi-level nature of FeFETs and can be expressed as follows: The FeFET only turns ON when its $V_{gs} > V_{th}$, therefore, if applying the same search sch but different store sto conditions results in different conducting states,i.e., $I_{sch,sto_a,i} \neq 0$ and $I_{sch,sto_b,i} = 0$, then the voltages corresponding to the search and store conditions must satisfy: $V_{sto_a} < V_{sch} < V_{sto_b}$. This stored threshold voltage relation must be satisfied when applying different search sch' condition, i.e., $I_{sch',sto_a,i} \geq I_{sch',sto_b,i}$. For example, the DMcurs values for the 2 FeFET under $Search11Store00$, $Search11Store01$, $Search00Store00$, $Search00Store01$ as shown in Fig. 4(e) results in a conflict, i.e., $V_{search,00} < V_{search,00}$. We refer to this as the third constraint.

The implemented CSP with above three constraints has many classical solution methods. Here, we choose Backtracking [30]

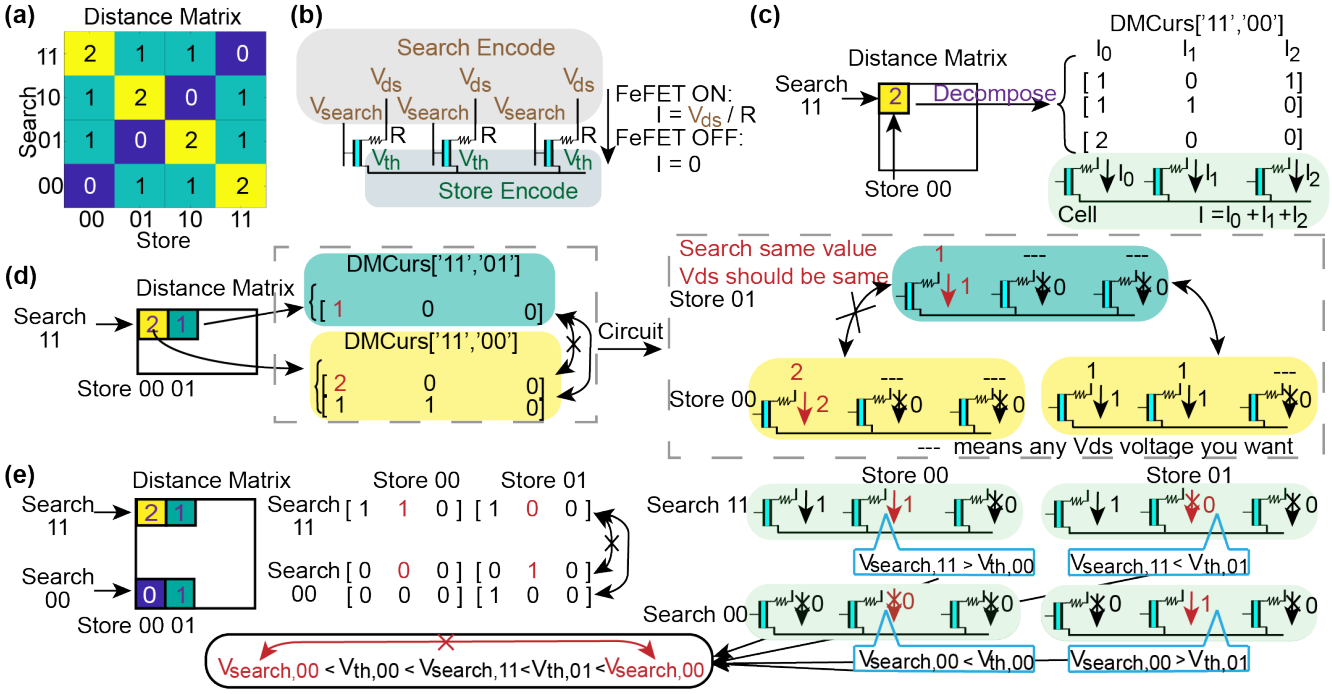

Fig. 4. (a) DM of 2-bit Hamming Distance. (b) Encoding with FeReX circuit. The stored encoding corresponds to programmed V_{th} values, while the search encoding corresponds to FeFET's V_{ds} and V_{gs} voltages. (c) DM element decomposition process based on the number of FeFETs in an AM cell. (d) and (e) The two constraint examples, where (d) for the same search voltage, the current of an FeFET must either be identical or 0, and (e) if $\text{FeFET}_{Search11,Store00,2}$ is ON, $\text{FeFET}_{Search11,Store01,2}$ is OFF, a conflict occurs if $\text{FeFET}_{Search00,Store00,2}$ is OFF and $\text{FeFET}_{Search00,Store01,2}$ is ON.

INPUT: The $M \times N$ Distance Matrix DM to be implemented by each cell, which includes K FeFETs, with a current range $CR = [C_1, C_2, \cdots C_n]$ allowed to flow through each FeFET
OUTPUT: *Feasible Region* or $False$

for *i from 0 to M-1* **do**
  for *j from 0 to N-1* **do**
    DMCurs[i, j] ← **DecomposeDM**(K, DM[i, j], CR)
  end
  Searchlines[i] ← **Backtracking**(DMCurs[i])
end
FeasibleRegion ← **AC3**(Searchlines)

if *Feasible Region not exist* **then**
  return False
end
return *Feasible Region*

Algorithm 1: FeReX Feasibility Detection Algorithm

Fig. 5. Encoding *Feasible Region* from algorithm. 1 to the store/search voltage configurations for a single FeFET device.

and AC3 [31], [32] to determine whether a set feasible FeFET currents under all sch and sto conditions exists, as illustrated in Alg. 1. The initial current set DMCurs is enumerated per the first constraint, and Backtracking and AC3 are utilized to effectively address the second and the third constraints, respectively. If the objective is to obtain all possible current sets, AC3 can be replaced by backtracking. The output of the algorithm is the *Feasible Region*, which consists of the filtered sets $DMCurs$ satisfying the three constraints.

Fig. 5 demonstrates the post processing of the output *Feasible Region* to obtain all the possible search and stored voltage encoding configurations for a single FeFET. For a feasible set DMCurs, during the stored sto encoding process, the numbers of ON states in all sto columns are counted and sorted. The sto columns with higher ranks correspond to lower V_{th} voltages. During the search sch encoding process, similarly, the numbers of OFF states in all sch rows are counted and sorted. The sch rows with higher ranks correspond to lower V_{search} voltages. The V_{ds} encoding corresponds to non-zero values in DMCurs.

Tab. II summarizes the encoding results for 2-bit Hamming Distance with the proposed FeReX circuit. FeReX iteratively increases the number of FeFETs within a cell, and determines that a 3FeFET3R cell structure is the optimal solution for

TABLE II
3FEFET3R 2BIT HAMMING DISTANCE ENCODING TABLE

	Store Encoding			Search Encoding					
	$V_{th,FET1}$	$V_{th,FET2}$	$V_{th,FET3}$	$V_{g,FET1}$	$V_{g,FET2}$	$V_{g,FET3}$	$V_{ds,FET1}$	$V_{ds,FET2}$	$V_{ds,FET3}$
"00"	Vt_2	Vt_2	Vt_0	Vs_2	Vs_2	Vs_0	V	V	V
"01"	Vt_2	Vt_0	Vt_2	Vs_1	Vs_0	Vs_2	$2V$	V	V
"10"	Vt_0	Vt_2	Vt_2	Vs_0	Vs_1	Vs_2	V	$2V$	V
"11"	Vt_1	Vt_1	Vt_1	Vs_1	Vs_1	Vs_1	V	V	$2V$

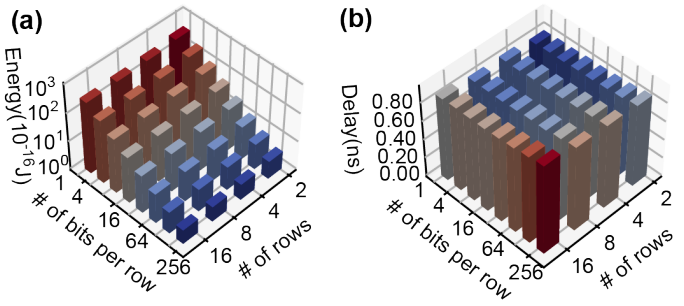

Fig. 6. Search energy and delay of FeReX: (a) Energy per bit and (b) delay with varying number of rows and dimensions.

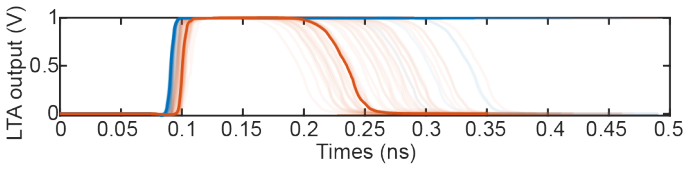

Fig. 7. Monte Carlo simulations considering device-to-device variations: FeReX achieves 90% accuracy in the worst search case of KNN workloads.

TABLE III
DATASETS (n: FEATURE SIZE, K: NUMBER OF CLASSES)

Dataset	n	K	Train Size	Test Size	Description
ISOLET	617	26	6,238	1,559	Voice Recognition [38]
UCIHAR	561	12	6,213	1,554	Physical Activity Monitoring [39]
MNIST	784	10	60,000	10,000	Handwritten Recongition [40]

the DM of 2-bit Hamming Distance. The FeFET is ON only if $V_{t_i} < V_{s_j}$, where $i < j, i, j \in \{0, 1, 2\}$. This encoding scheme has also been extended to other distance functions such as multi-bit Manhattan and multi-bit Euclidean. We leverage encoding of multi-bit Manhattan and multi-bit Euclidean in Sec. IV-B for benchmarking.

IV. EVALUATION & BENCHMARKING

In this section, we evaluate the FeReX using Cadence Virtuoso in terms of accuracy, robustness, and power consumption. The Preisach FeFET model [33] was adopted for FeFETs, while the 45nm PTM model [34] was used for all MOSFETs. Wiring parasitics for the 45nm technology node were extracted from DESTINY [35]. The operational amplifier (op-amp) was based on the design from [36] and scaled down to 45nm technology.

A. Array Evaluation

Fig. 6(a) demonstrates that increasing the number of rows in the FeReX can reduce the average energy consumption per bit, since the power consumption of LTA grows insignificantly as the number of rows increases. The search delay consists of two parts. About 60% of the total delay comes from ScL voltage stabilization associated with the op-amp, which is constrained by the op-amp's slew rate. The remaining delay associates with the LTA circuitry. As shown in Fig. 6(b), the total delay increases gradually as the FeReX array scales.

To further validate the effectiveness of the proposed FeReX, we conduct Monte Carlo (MC) simulation in the context of KNN, by taking device-to-device variation into account. Then, we benchmark FeReX with the vector-symbolic architecture (VSA) framework [37], also known as the hyperdimensional computing (HDC). Fig. 7 illustrates the MC simulations of FeReX with 100 runs. The device-to-device variation for the FeFET threshold voltage was set to 54mV [20], and the resistance variation for the 1FeFET1R structure was extracted from fabricated data [22], set to 8%. The FeReX array level results demonstrate 90% search accuracy when comparing the stored vectors with Hamming distances 5 and 6 to the query, representing the most challenging search casesof KNN when executing MNIST. This performance results in only a 0.6% accuracy degradation compared to the software-based implementation.

B. Application Benchmarking

we briefly discuss the advantages of HDC benchmarking and its algorithmic flow. In HDC, low dimensional features are initially projected to high dimensional representations randomly, enabling *holographicness* across the high dimensional feature vectors. HDC is pre-defined through a set of transparent operations, and due to its holographicness, it has been reported to be robust against hardware noise [41].

The algorithmic flow of HDC can be categorized into three steps: first, data is projected to high dimension, as mentioned above. Second, single-pass training is performed, where the encoded high-dimensional vectors of a certain class are aggregated. Iterative training are conducted for higher algorithmic accuracy. Finally, during the inference phase of classification, the predicted class vector that has closest distance to the query vector is output using the configured FeReX distance function.

Here, we benchmark the proposed FeReX in the context of HDC with Nvidia 3090 GPU [42] over three large-scale datasets given in Tab. III. By extracting the latency of the inference operations through *Pytorch Profiler* package, the energy is obtained with the Nvidia System Management Interface. Fig. 8(a) shows the accuracy of the reconfigurable search engine. Conventional CIM-based HDC accelerator implements Hamming distance,

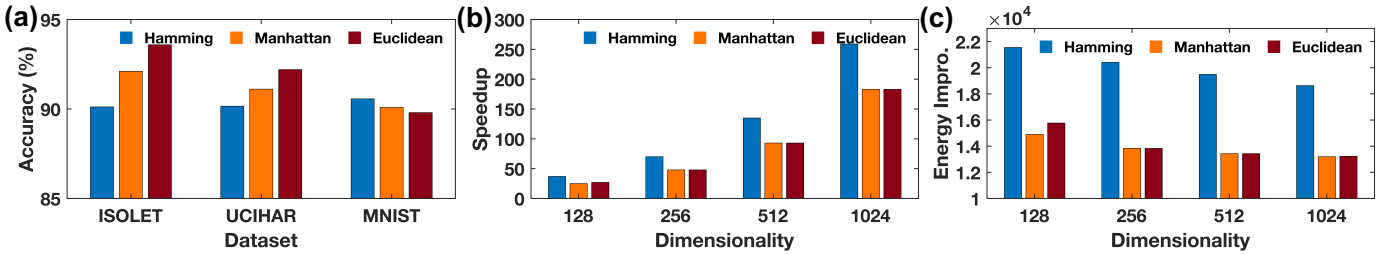

Fig. 8. (a) Classification accuracy with different FeReX distance metrics. (b) Computation speedup and (c) energy efficiency improvement over GPU implementation.

yet different distance metrics may result in better accuracy across different datasets. Fig. 8(b) and (c) show the efficiency of the proposed FeReX, showcasing up to 250x speedup and 10^4 energy improvement over the GPU implementation.

V. Conclusion

In this paper, we propose FeReX, a FeFET-based AM for reconfigurable distance NN search. Based on derived FeFET device and circuit constraints, FeReX filters and encodes feasible search and stored voltage configurations to implement a distance matrix of the target distance function by addressing the constraint satisfaction problem. Evaluations at array level validate the functionality and efficiency of the proposed FeReX, and benchmarking results illustrate the improvement of FeReX implementation over GPU. To the best of our knowledge, this is the first NVM based AM with reconfigurable search distance function, which will pave the way towards reconfigurable AM designs for broader ranges of emerging applications.

Acknowledgements

This work was supported in part by National Key R&D Program of China (2022YFB4400300), National Natural Science Foundation of China (62104213, 92164203, 62122005), Zhejiang Provincial Natural Science Foundation (LD21F040003, LQ21F040006), the Research Grant Council of HKSAR (27210321), the Croucher Foundation and the ACCESS—AI Chip Center for Emerging Smart Systems, sponsored by InnoHK funding, Hong Kong SAR. Liu was supported by CoCoSys, one of seven centers in JUMP2.0, a Semiconductor Research Corporation (SRC) program sponsored by DARPA. Imani was supported in part by National Science Foundation #2127780, #2319198, #2321840 and #2312517, Office of Naval Research #N00014-21-1-2225 and #N00014-22-1-2067, the Air Force Office of Scientific Research #FA9550-22-1-0253.

References

[1] M. Li *et al.*, "Imars: An in-memory-computing architecture for recommendation systems," in *ACM/IEEE DAC*, 2022, pp. 463–468.
[2] Y. Wei *et al.*, "Imga: Efficient in-memory graph convolution network aggregation with data flow optimizations," *IEEE TCAD*, 2023.
[3] Z. Yan *et al.*, "Swim: Selective write-verify for computing-in-memory neural accelerators," in *IEEE DAC*, 2022, pp. 277–282.
[4] X. Chen *et al.*, "Accelerating graph-connected component computation with emerging processing-in-memory architecture," *TCAD*, vol. 41(12), pp. 5333–5342, 2022.
[5] Z. Yan *et al.*, "Computing-in-memory neural network accelerators for safety-critical systems: Can small device variations be disastrous?" in *IEEE ICCAD*, 2022, pp. 1–9.
[6] X. S. Hu *et al.*, "In-memory computing with associative memories: A cross-layer perspective," in *IEEE IEDM*, 2021, pp. 25–2.
[7] A. S. Lele *et al.*, "A heterogeneous rram in-memory and sram near-memory soc for fused frame and event-based target identification and tracking," *IEEE JSSC*, 2023.
[8] X. Yin *et al.*, "Ferroelectric ternary content addressable memories for energy-efficient associative search," *IEEE TCAD*, vol. 42, no. 4, pp. 1099–1112, 2022.
[9] X. Peng *et al.*, "Dnn+ neurosim v2. 0: An end-to-end benchmarking framework for compute-in-memory accelerators for on-chip training," *IEEE TCAD*, 2020.
[10] J. Cai *et al.*, "Energy efficient data search design and optimization based on a compact ferroelectric fet content addressable memory," in *2022 DAC*, pp. 751–756.
[11] X. Wang *et al.*, "Triangle counting accelerations: From algorithm to in-memory computing architecture," *IEEE TC*, vol. 71, no. 10, pp. 2462–2472, 2021.
[12] X. Yin *et al.*, "Fecam: A universal compact digital and analog content addressable memory using ferroelectric," *IEEE TED*, vol. 67, no. 7, pp. 2785–2792, 2020.
[13] K. Ni *et al.*, "Ferroelectric ternary content-addressable memory for one-shot learning," *Nature Electronics*, 2019.
[14] H. Li *et al.*, "Sapiens: A 64-kb rram-based non-volatile associative memory for one-shot learning and inference at the edge," *IEEE TED*, 2021.
[15] A. Kazemi *et al.*, "Achieving software-equivalent accuracy for hyperdimensional computing with ferroelectric-based in-memory computing," *Scientific reports*, 2022.
[16] L. Liu *et al.*, "A reconfigurable fefet content addressable memory for multi-state hamming distance," *IEEE TCAS-I*, 2023.
[17] H. Xu *et al.*, "On the challenges and design mitigations of single transistor ferroelectric content addressable memory," *IEEE EDL*, 2023.
[18] A. Kazemi *et al.*, "Fefet multi-bit content-addressable memories for in-memory nearest neighbor search," *IEEE TC*, 2021.
[19] T. Böscke *et al.*, "Ferroelectricity in hafnium oxide: Cmos compatible ferroelectric field effect transistors," in *IEEE IEDM*, 2011.
[20] T. Soliman *et al.*, "Ultra-low power flexible precision fefet based analog in-memory computing," in *IEDM*. IEEE, 2020.
[21] X. Yin *et al.*, "An ultracompact single-ferroelectric field-effect transistor binary and multibit associative search engine," *Advanced Intelligent Systems*, 2023.
[22] D. Saito *et al.*, "Analog in-memory computing in fefet-based 1t1r array for edge ai applications," in *IEEE Symp. on VLSI Tech.*, 2021.
[23] G. Karunaratne *et al.*, "In-memory hyperdimensional computing," *Nature Electronics*, vol. 3, no. 6, 2020.
[24] C. Li *et al.*, "A scalable design of multi-bit ferroelectric content addressable memory for data-centric computing," in *IEEE IEDM*, 2020.
[25] S. Shou *et al.*, "See-mcam: Scalable multi-bit fefet content addressable memories for energy efficient associative search," in *IEEE/ACM ICCAD*, 2023.
[26] Q. Huang *et al.*, "Fefet based in-memory hyperdimensional encoding design," *IEEE TCAD*, 2023.
[27] P.-Y. Chen *et al.*, "Neurosim: A circuit-level macro model for benchmarking neuro-inspired architectures in online learning," *IEEE TCAD*, 2018.
[28] K. Ni *et al.*, "Write disturb in ferroelectric fets and its implication for 1t-fefet and memory arrays," *IEEE EDL*, 2018.
[29] C.-K. Liu *et al.*, "Cosime: Fefet based associative memory for in-memory cosine similarity search," in *IEEE/ACM ICCAD*, 2022.
[30] J. R. Bitner *et al.*, "Backtrack programming techniques," *Communications of the ACM*, 1975.
[31] A. K. Mackworth, "Consistency in networks of relations," *Artificial intelligence*, 1977.
[32] R. Soto *et al.*, "A hybrid ac3-tabu search algorithm for solving sudoku puzzles," *Expert Systems with Applications*, 2013.
[33] K. Ni *et al.*, "A circuit compatible accurate compact model for ferroelectric-fets," in *IEEE Symp. on VLSI Tech.*, 2018.
[34] R. Vattikonda *et al.*, "Modeling and minimization of pmos nbti effect for robust nanometer design," in *IEEE DAC*, 2006.
[35] M. Poremba *et al.*, "Destiny: A tool for modeling emerging 3d nvm and edram caches," in *IEEE DATE*, 2015.
[36] B. H. Kassiri *et al.*, "Slew-rate enhancement for a single-ended low-power two-stage amplifier," in *IEEE ISCAS*, 2013.
[37] D. Kleyko *et al.*, "Vector symbolic architectures as a computing framework for emerging hardware," *Proceedings of the IEEE*, 2022.
[38] "Uci machine learning repository," http://archive.ics.uci.edu/ml/datasets/ISOLET.
[39] D. Anguita *et al.*, "Human activity recognition on smartphones using a multiclass hardware-friendly support vector machine," in *IWAAL*. Springer, 2012.
[40] Y. LeCun *et al.*, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, 1998.
[41] A. Hernández-Cano *et al.*, "Reghd: Robust and efficient regression in hyper-dimensional learning system," in *IEEE DAC*, 2021, pp. 7–12.
[42] A. Hernández-Cano *et al.*, "Onlinehd: Robust, efficient, and single-pass online learning using hyperdimensional system," in *IEEE DATE*, 2021.