RESEARCH ARTICLE | FEBRUARY 14 2023

Benchmarking energy consumption and latency for neuromorphic computing in condensed matter and particle physics

Special Collection: 2023 Papers with Best Practices in Data Sharing and Comprehensive Background Review

Dominique J. Kösters ; Bryan A. Kortman ; Irem Boybat ; Elena Ferro ; Sagar Dolas;
Roberto Ruiz de Austri; Johan Kwisthout; Hans Hilgenkamp; Theo Rasing; Heike Riel; Abu Sebastian;
Sascha Caron; Johan H. Mentink

APL Mach. Learn. 1, 016101 (2023)
https://doi.org/10.1063/5.0116699

Benchmarking energy consumption and latency for neuromorphic computing in condensed matter and particle physics

Cite as: APL Mach. Learn. 1, 016101 (2023); doi: 10.1063/5.0116699
Submitted: 29 July 2022 • Accepted: 3 November 2022 •
Published Online: 14 February 2023

View Online Export Citation CrossMark

Dominique J. Kösters,[1,2,3] Bryan A. Kortman,[1,2,4] Irem Boybat,[3] Elena Ferro,[3,5] Sagar Dolas,[6] Roberto Ruiz de Austri,[7] Johan Kwisthout,[8] Hans Hilgenkamp,[1,9] Theo Rasing,[2] Heike Riel,[3] Abu Sebastian,[3] Sascha Caron,[4,10] and Johan H. Mentink[2,a)]

AFFILIATIONS

[1]Faculty of Science and Technology, University of Twente, P.O. Box 217, 7500 AE Enschede, The Netherlands
[2]Institute for Molecules and Materials, Radboud University, Heyendaalseweg 135, 6525 AJ Nijmegen, The Netherlands
[3]IBM Research Europe-Zürich, Säumerstrasse 4, 8803 Rüschlikon, Switzerland
[4]Nikhef, P.O. Box 41882, 1098 XG Amsterdam, The Netherlands
[5]Eidgenössische Technische Hochschule Zürich, Department of Information Technology and Electrical Engineering, Gloriastrasse 35, 8092 Zürich, Switzerland
[6]SURF Cooperation, Innovation Team, Moreelespark 48, 3511 EP Utrecht, The Netherlands
[7]Instituto de Física Corpuscular, Parc Científic UV, University of Valencia-CSIC, c/Catedrático José Beltrán 2, E-46980 Paterna, Spain
[8]Donders Institute for Brain, Cognition and Behaviour, Radboud University, P.O. Box 9104, 6500 HE Nijmegen, The Netherlands
[9]MESA+ Institute for Nanotechnology, University of Twente, P.O. Box 217, 7500 AE Enschede, The Netherlands
[10]Institute for Mathematics, Astrophysics and Particle Physics, Radboud University, Heyendaalseweg 135, 6525 AJ Nijmegen, The Netherlands

[a)]**Author to whom correspondence should be addressed:** j.mentink@science.ru.nl

ABSTRACT

The massive use of artificial neural networks (ANNs), increasingly popular in many areas of scientific computing, rapidly increases the energy consumption of modern high-performance computing systems. An appealing and possibly more sustainable alternative is provided by novel neuromorphic paradigms, which directly implement ANNs in hardware. However, little is known about the actual benefits of running ANNs on neuromorphic hardware for use cases in scientific computing. Here, we present a methodology for measuring the energy cost and compute time for inference tasks with ANNs on conventional hardware. In addition, we have designed an architecture for these tasks and estimate the same metrics based on a state-of-the-art analog in-memory computing (AIMC) platform, one of the key paradigms in neuromorphic computing. Both methodologies are compared for a use case in quantum many-body physics in two-dimensional condensed matter systems and for anomaly detection at 40 MHz rates at the Large Hadron Collider in particle physics. We find that AIMC can achieve up to one order of magnitude shorter computation times than conventional hardware at an energy cost that is up to three orders of magnitude smaller. This suggests great potential for faster and more sustainable scientific computing with neuromorphic hardware.

I. INTRODUCTION

The energy demand of modern high-performance computing systems is currently rapidly rising owing to the massive use of artificial neural networks (ANNs).[1] Moreover, for several of the most challenging compute tasks in scientific computing, applications of ANNs offer competitive advantages over standard algorithms. This includes various examples in condensed matter physics[2–4] and particle physics,[5–7] which feature data rates as high as 1 Pb/s in bursts separated by just 25 ns.[8] Such workloads are very challenging for conventional hardware since the corresponding energy consumption is simply too high, even for inference tasks with pre-trained networks.

Neuromorphic hardware offers great potential as an accelerator for such highly demanding compute tasks. For example, spiking neural networks are considered advantageous for optimization problems[9–12] and have also been recently applied to entangled quantum states.[13] Another efficient realization of ANNs involves physically instantiating the synaptic weights in memory devices and exploiting the physical attributes of these memory devices to implement the ANN in hardware.[14–16] This approach, typically referred to as analog in-memory computing (AIMC), would obviate the need to shuttle millions of synaptic weights between the memory and processing units and could lead to significant gains in energy efficiency and latency. However, rather little is known about the actual benefits of running ANNs on AIMC for concrete physics use cases. Moreover, physics users often do not even consider the energy cost of complete ANN workloads as a relevant figure of merit.

In this Letter, we aim to assess the potential of AIMC accelerators for ANN-based use cases in condensed matter and particle physics by benchmarking them against conventional hardware implementations of the ANNs. To this end, we develop a generic methodology for measuring the energy cost and compute time for inference tasks with ANNs on central processing unit (CPU) and graphics processing unit (GPU) hardware. In addition, we have designed an architecture to estimate these metrics for inference tasks on a Mixed-Precision Analog In-Memory Computing (MP-AIMC) platform. The scientific use cases chosen feature computation of quantum many-body states in two dimensions[17,18] and the 40 MHz challenge for anomaly detection at the Large Hadron Collider (LHC) in particle physics.[19,20] By comparing the measurements on conventional hardware against the estimations for MP-AIMC, we find that the latter can reach up to an order of magnitude shorter compute time and down to three orders of magnitude lower energy cost.

II. SCIENTIFIC USE CASES

A. Condensed matter physics

Understanding the effect of correlations on the properties of quantum many-body systems is one of the most fascinating research fields in condensed matter physics. Recently, a new method for the simulation of quantum many-body systems has been pioneered, inspired by machine learning.[17] In this approach, the many-body wave function is approximated by an ANN and the quantum states generated this way are termed neural-network quantum states (NQSs). Already the simplest network, the Restricted Boltzmann Machine (RBM), was found to give competitive advantages over conventional methods, in particular, in two dimensions (2D) for which quantum correlations are strongest.[17,18,21,22] Despite this potential, the method suffers from long training times when applied to large systems. Moreover, extracting observables from an already trained network takes up a large part of the computational effort.

For this use case, we consider NQS for the 2D antiferromagnetic Heisenberg model on a square lattice. The Hamiltonian is defined as

$$\hat{H} = J \sum_{\langle i,j \rangle} \hat{S}_i \cdot \hat{S}_j, \tag{1}$$

where J is the exchange constant, $\hat{S}_i$ is the quantum spin operators for $S = 1/2$, and the sum runs over the nearest neighbors. In the context of NQS, an inference task is defined by the evaluation of the RBM wave function, which for translation invariant systems can be written as

$$\psi(s) = \prod_{i=1}^{\alpha N} 2\cosh\big([Ws + b]_i\big), \tag{2}$$

where α is the ratio between the number of spins and hidden layer nodes, N is the number of spins in the lattice, W is a matrix of dimensions $\alpha N \times N$ with the weights, b is a vector of dimension αN with the biases, and s represents the input spin configuration, which is encoded in a binary tuple (column vector) $s = (s_1, \ldots, s_N)^T$, with $s_j = \pm 1$. A schematic representation of the RBM is shown in Fig. 1. For large systems, $\psi(s)$ is evaluated only for a subset of the 2^N possible states by Monte Carlo sampling. For our assessment, it is sufficient to consider a 4×4 system for which the input dataset is defined by all possible states with zero magnetization.

In order to obtain ground state properties, the network is trained by minimizing the energy $E = \langle \psi | \hat{H} | \psi \rangle / \langle \psi | \psi \rangle$ using the stochastic reconfiguration method,[23] while a closely related training procedure is obtained for quantum dynamics based on the time-dependent variational principle.[17,24] In either case, the training itself

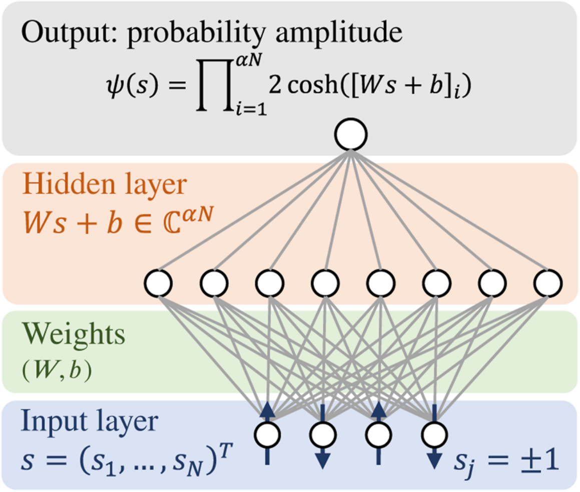
Output: probability amplitude
$\psi(s) = \prod_{i=1}^{\alpha N} 2\cosh([Ws + b]_i)$
Hidden layer
$Ws + b \in \mathbb{C}^{\alpha N}$
Weights
(W, b)
Input layer
$s = (s_1, \ldots, s_N)^T$
$s_j = \pm 1$

FIG. 1. Schematic representation of the RBM. The hidden layer size is αN, with α being a positive integer (in this schematic representation, $N = 4$ and $\alpha = 2$). Each hidden neuron features a hyperbolic cosine activation. Their outputs are multiplied to produce the network output.

relies on the repeated evaluation of $\psi(s)$ at fixed RBM parameters before updates are computed. Moreover, evaluation of observables exclusively relies on inference. Therefore, we focus on the inference procedure alone and take already trained weights for the RBM ground state.[18] In this case, both the weights and biases can be chosen to be real valued.

B. Particle physics

One of the most important challenges in High Energy Physics (HEP) today is to find rare new physics signals among an abundance of Standard Model (SM) proton–proton collisions. This use case[20] uses Deep Learning (DL) techniques to find anomalous signals at the LHC among many SM events, also known as anomaly detection. The event rate from which events must be selected at the ATLAS detector is 40 MHz, brought down to the final collection rate of 300 Hz using a three staged trigger system. The network in this use case is designed to run on the level-1 trigger system responsible for reducing the rate to less than 75 kHz. Therefore, very low latency is required to ensure the network can keep up with the proton–proton collision rate.

The network in this use case is designed to detect events that are forbidden in the SM. This is achieved by a one-class Deep Support Vector Data Description (Deep SVDD) approach, trained to map every input of the Deep SVDD onto a predefined multidimensional point. The distance to this point is then regarded as the final anomaly score. A schematic representation of the SVDD is shown in Fig. 2. The models are trained under the assumption that all SM data fall inside the predetermined manifold. During the testing, the beyond the Standard Model (BSM) data will fall outside this manifold. Several Deep SVDD networks are combined into an ensemble to maximize the efficiency. The networks are fully connected networks that output a constant vector for every input. The loss is defined as

$$S(x) = \left[O_n^z - \text{model}(x)\right]^2. \tag{3}$$

The model maps the input x to the same tensor shape as the vector O, with z and n referring to the number of elements and the scalar value, respectively. The activation function of the hidden layers is an exponential linear unit (ELU). In this use case, we employ an ensemble of 63 networks in total. They contain the same network structure but different combinations of values [5, 8, 13, 21, 34, 55, 89, 144, 233] for z and [0, 1, 2, 3, 4, 10, 25] for n.

The data used to test and train this use case consist of five data records.[8] One record containing a mixture of SM processes and four separate records of BSM processes. The data are generated using Pythia 8.240[25] using a collision energy of $\sqrt{s} = 13$ TeV. The BSM processes were $A \to 4l$, $LQ \to b\tau$, $H \to \tau\tau$, and $H^{\pm} \to \tau\nu$. The detector response is modeled with DELPHES 3.3.2.[26] Input variables available for each event are the p_T, η, and ϕ values of the missing energy, four electrons, four muons, and ten jets.

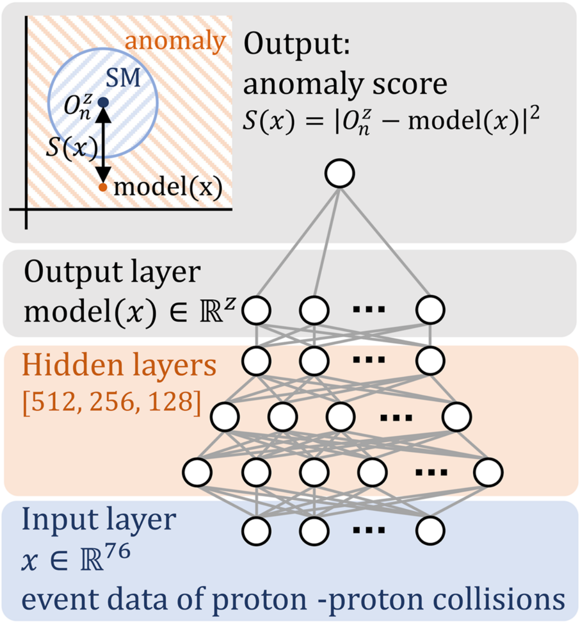

FIG. 2. A schematic representation of the Deep SVDD model. The input is the event data of proton–proton collisions, and the output is a z-dimensional vector denoted model (x). An anomaly score can be given to the event by comparing the model output with a target (O_n^z). In this case, the model output is far from the target, and therefore, the event can be considered an anomaly.

III. METHODS

A. Energy-measurement methodology CPU and GPU

Energy measurements were performed on the Emergency Smart Computing (ESC) cluster within Innovation Labs at SURF. The inference of the networks is performed on a dual socket system with two Intel Xeon Gold CPUs and a NVIDIA V100 GPU. The measurements are performed using the Energy Aware Runtime (EAR) software package[27] and provide an energy management framework for experimental computing. The inference jobs are monitored by measuring the DC node power using a Baseboard Management Controller (BMC), which is a specialized service processor that monitors the physical state of the hardware device using sensors and communicates with the system administrator through an independent connection. The BMC is part of the Intelligent Platform Management Interface (IPMI), which is a standardized message-based hardware management interface.

EAR is currently implemented to give the metrics of an entire submitted job on a specific node. The energy metric used employs the DC node energy $E_{\text{DC}}^{\text{node}}$. The metrics presented below are the average energy per inferred sample, defined as

$$E_{\text{sample}} = \frac{E_{\text{DC}}^{\text{node}}}{N}, \tag{4}$$

where N is the number of inferred samples, and the throughput T is defined as

$$T = \frac{N}{\Delta t}, \tag{5}$$

where Δt is the elapsed time. An effective latency could be derived from the throughput by taking the inverse, i.e., $L = T^{-1}$.

The samples were supplied in batches to the ANNs. The batch size was chosen to minimize E_{sample} by doing a sweep search. The most energy efficient batch size was in these cases also the batch size with the highest throughput. To ensure that initialization processes are insignificant in terms of energy and time, the total computation time of inference is forced to be larger than 99% by increasing the number of inference steps.

B. Energy estimation for mixed-precision analog in-memory computing hardware

In this section, we present a dedicated MP-AIMC hardware design, which can support both the physics use cases described earlier. Although originating from vastly different fields of physics, the operations necessary to compute the wave function $\psi(S)$ and the anomaly score are similar, and therefore the two use cases are easily combined in one design.

AIMC hardware is based on memory crossbar arrays with stationary weights, implementing matrix–vector multiplication (MVM) directly in hardware. The proposed MP-AIMC architecture, shown in Fig. 3, is composed of four analog tiles and a Digital Processing Unit (DPU). The analog tile is assumed to be similar to a phase-change memory-based AIMC tile designed and fabricated in a 14 nm CMOS technology node.[28,29] The DPU components are also estimated based on circuit designs at the same technology node.

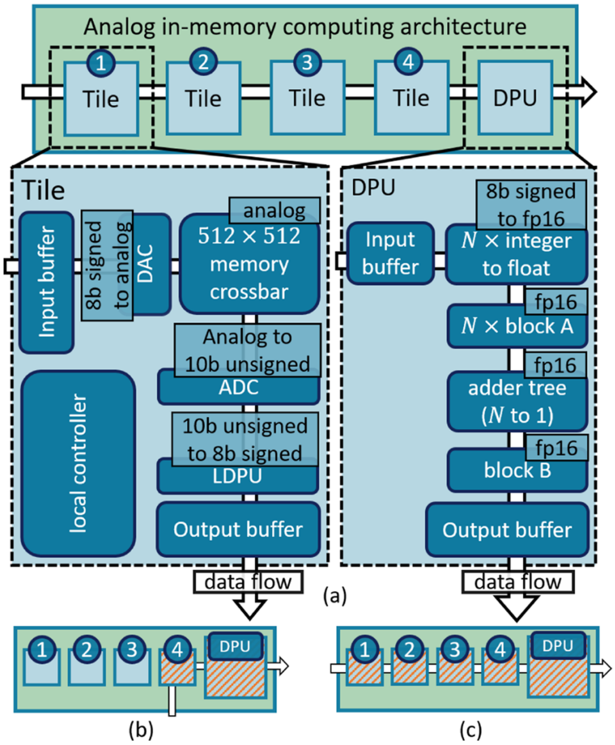

FIG. 3. Proposed MP-AIMC architecture. (a) Block diagram of the proposed AIMC-based architecture showing four identical tiles and the DPU. It includes a zoom-in of one tile and a zoom-in of the DPU where the corresponding data types are mentioned per block. Included in blocks A and B are a hard-wired look-up table, floating point multiplier, and floating point adder. (b) The utilization of the proposed AIMC-based architecture for the condensed matter physics use case, where only tile 4 and the DPU are utilized. (c) The utilization for the particle physics use case, where all four tiles and the DPU are utilized.

The four tiles are identical and consist of a local controller, 512 Digital-to-Analog Converters (DACs), a 512 × 512 memory crossbar with Phase Change Memory (PCM) devices, 512 Analog-to-Digital Converters (ADCs), and a Local Digital Processing Unit (LDPU).

The DAC converts an 8-bit signed integer to a voltage pulse with fixed height (negative and positive numbers have a different fixed height) where the width of the pulse corresponds to the value of the integer. The DACs are connected to the wordlines of the memory crossbar. The crossbar evaluates the vector matrix multiplication by accumulating current along the bitlines according to Ohm's law and Kirchoff's law. The ADCs are variants of the current-controlled oscillator-based ADCs, as described in Ref. 28. The instantaneous current flowing through the bitlines is digitized and accumulated in a digital counter (which is also part of the ADC) during the application of the input pulse. The ADC outputs two 10-bit unsigned integers, one integer corresponding to positive values and the other corresponding to negative values. Due to the compact design of the ADCs, it is possible to operate all 512 ADCs in parallel. We emphasize that the input and output buffers for the tiles and DPU are considered in the estimation. Moreover, both LDPU and DPU consist of internal stages that are divided by registers, which are also included. The combined latency of the DACs, ADCs, and the memory crossbar is estimated to be 40 ns.

The LDPU implements the affine scaling, addition of biases, and optionally the Rectified Linear Unit (ReLU) activation function. Due to practical considerations, the Deep SVDD implementation on the MP-AIMC architecture employs a ReLU instead of an ELU activation function. Both activation functions result in a similar accuracy. The LDPU and all the components used inside the DPU are synthesized to verify that both timing and area constraints are met and custom designed for relatively low-precision operations.

The additional DPU takes the network output and computes $\psi(s)$ for NQS, and for the SVDD, it evaluates the Euclidean distance between the target, O_n^z, and the network output, which can be viewed as the anomaly score. This evaluation can be done with the use of look-up tables for activation functions and an adder tree for both use cases. For NQS, only one analog tile and the DPU are utilized [shown in Fig. 3(c)], while for the SVDD application, all four tiles and DPU are utilized [Fig. 3(b)].

Due to the stationary nature of the weights in the proposed MP-AIMC architecture, this design is well suited for a data-flow in a pipeline fashion. This means that the data-flow can be divided into several sequential pipeline stages. The throughput is then limited by the slowest pipeline stage, which in this design is estimated to be 50 ns, and is independent of network size as long as the individual layers fit on the 512 × 512 crossbar arrays. When running the workloads of the two use cases with maximum load (entire 512 × 512 crossbar is utilized), the power consumption of the crossbar (the memory array and the peripheral circuitry, including the data converters) is estimated to be 0.13 W, where the peripheral circuitry contributes about 90% of the power consumption. The average power consumption of the LDPU and DPU after synthesis is estimated to be 0.33 and 0.18 W, respectively. In this case, the average power consumption for the computational and particle physics use cases is 0.63 and 2.03 W, respectively. Note that, in the former case, just one tile and the DPU are used, whereas in the latter case, all four tiles and the DPU are employed. Energy consumption of communication between tiles and between the final tile and the DPU is included in the energy estimation.

IV. RESULTS

A. Condensed matter physics

Figure 4(a) shows the average energy of inference, and Fig. 4(b) shows the throughput, both as a function of the hidden layer density α. For the NVIDIA GPU, the energy consumption (throughput) increases (decreases) with hidden layer size and by increasing the data precision from floating point 32 to floating point 64, the two commonly used precision settings in the field. For both metrics, the GPU outperforms the CPU up to a factor 10. The dual socket Intel Xeon Gold CPU features a non-monotonic dependence on network size and floating point precision. This is attributed to irregular utilization of the cores (40 in total). Forcing execution on a single core recovers the expected linear scaling with hidden layer density as we have confirmed by independent measurements (data not shown).

The energy and throughput estimates from the proposed MP-AIMC architecture are also shown in Figs. 4(a) and 4(b) (dashed lines). The estimated energy use of this architecture is up to a factor 10^3 lower than the CPU and GPU. We observe that the MP-AIMC design yields a throughput comparable to CPU. However, GPU outperforms the MP-AIMC throughput up to a factor 7 for $\alpha \leq 4$. For large networks, the GPU throughput reduces, whereas MP-AIMC remains flat by design, given that the hidden layer fits into a single MP-AIMC tile. This suggests that, for even larger networks, MP-AIMC will also have an advantage for the throughput. In addition, we note that unlike both the CPU and the GPU, the proposed MP-AIMC architecture features no parallelization. The throughput of the MP-AIMC architecture naturally increases if multiple networks are run in parallel using more tiles and DPU resources at the expense of an increase in area of the chip.

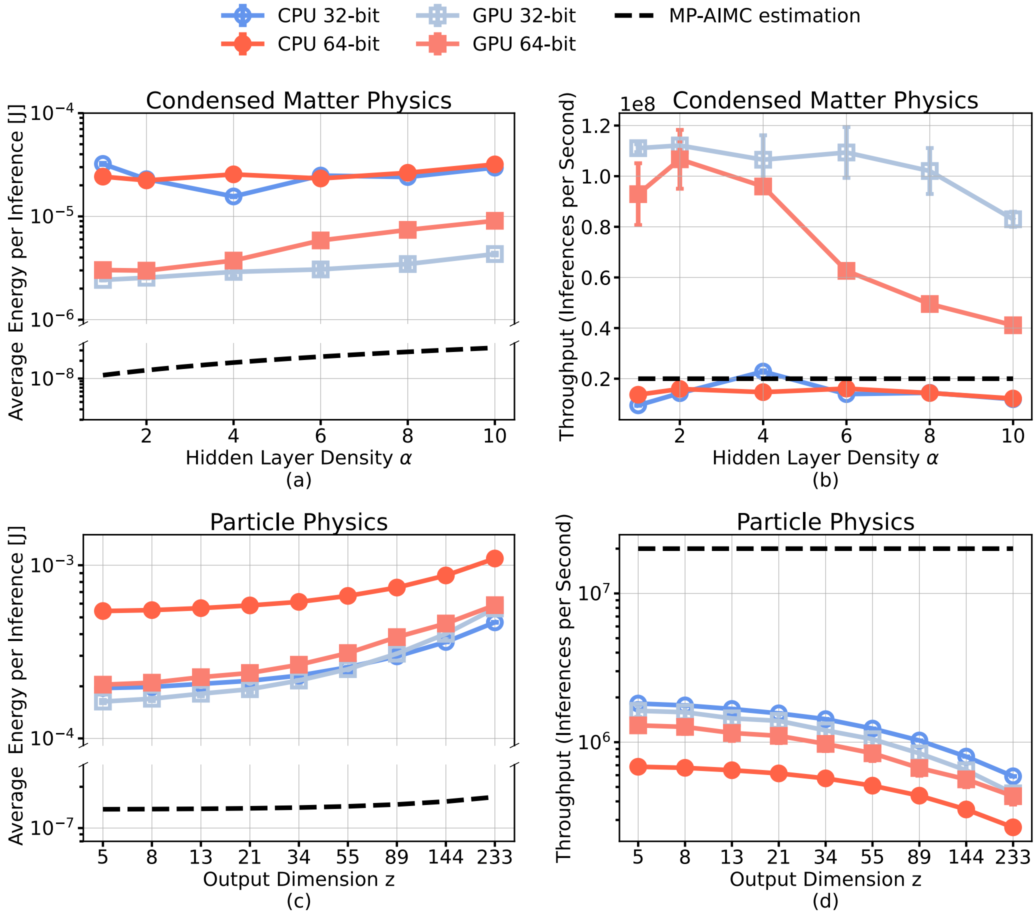

FIG. 4. Results of benchmarks on CPU (Intel Xeon Gold dual socket) and GPU (NVIDIA V100) and estimations for the proposed AIMC architecture. (a) The average energy per inference (or average energy per state) in joule and (b) the throughput inferences per second (or states per second) for the condensed matter physics use case. (c) The average energy per inference (or average energy per event) in joule and (d) the throughput inferences per second (or events per second) for the particle physics use case. The measurements on the CPU and GPU are repeated ten times, and the standard deviation is included in the plots. The batch size used for the condensed matter physics use case is equal to all states with zero magnetization (12 870), and the batch size used for the particle physics use case is 10^6. Horizontal axes indicate a measure defining the network complexity/size: this is the hidden layer density α for the condensed matter physics use case and the output dimension z for the particle physics use case.

B. Particle physics

Figure 4(c) shows the average energy consumption of a batch of inferred events, and Fig. 4(d) shows the throughput as a function of the individual network design of a single SVDD in the SVDD ensemble, labeled with z, the dimension of the manifold. Distinct from the condensed matter physics use case, GPU and CPU score very similar results for both metrics, in particular, for single precision. Interestingly, in this case, the CPU outperforms the GPU for the energy cost at the largest network sizes ($z > 144$). In addition, for the throughput, the performance difference between CPU and GPU is small. GPU is found to be superior to the CPU for double precision, while for single precision, again CPU performs better, now for all network sizes. This nontrivial behavior is attributed to the computational cost stemming from the distance calculation [Eq. (3)], which in turn determines the anomaly score, which we confirmed by leaving out the distance calculation in the code.

Figures 4(c) and 4(d) also include the energy estimation of the proposed MP-AIMC architecture, which is observed to be up to a factor of 10^3 more efficient than CPU and GPU. By the pipelined dataflow, the MP-AIMC throughput is the same as for the condensed matter physics use case. However, on GPU and CPU, the inference with the more complex and deeper structure of the SVDD requires more computational efforts than the evaluation of the shallow RBM. As a result, MP-AIMC can yield over a factor 20 faster throughput than CPU and GPU at the largest SVDD output dimensions.

V. CONCLUSION

We have presented a methodology to measure energy cost and compute time on CPU and GPU for inference tasks based on ANNs. By applying this methodology to NQS for quantum many-body systems in condensed matter physics and SVDD networks for anomaly detection in particle physics, we found that benefits of GPU, as compared to CPU, for energy efficiency and throughput strongly depend on the ANN architecture and non-MVM operations. In particular, CPU can outperform GPU even for the largest networks considered. Therefore, energy benchmarks are always important, especially when working with unorthodox experimental ANN-based algorithms.

Furthermore, we have proposed a dedicated MP-AIMC architecture capable of implementing both physics use cases, based on which the energy consumption and throughput can be estimated. By comparing the measured energy on CPU and GPU with the energy estimations for MP-AIMC, it is found that the latter improves the energy efficiency up to a factor 10^3 for both the condensed matter physics and particle physics use case. The throughput is flat as a function of the network size in the proposed MP-AIMC architecture, as long as the network fits in the 512×512 crossbar array. For the relatively small networks used for NQS tested for the condensed matter physics use case, this yields a MP-AIMC throughput comparable with that of the CPU, whereas GPU throughput is up to factor 7 higher. Importantly, for the larger SVDD networks in the particle physics use case, the MP-AIMC throughput is always higher, over a factor 20 for the largest network tested.

The benchmarks performed suggest great potential for neuromorphic accelerators based on MP-AIMC. A key challenge associated with AIMC is computational imprecision.[30] Yet, solutions on device,[31] unit cell and circuit,[32] and algorithmic level[33] have been shown to be effective in compensating for the low-precision analog computing. It is foreseen that similar approaches can be effective for the networks presented in this work and consider studying the impact of precision as a very interesting topic for future work, which should include both nonidealities of PCM and those of peripheral circuitry. Moreover, future work may focus on next-generation cross-bar structures and parallel architectures that can further improve the throughput. Combined with the fundamentally flat scaling of the compute time with the network size, this suggests great potential for scientific workloads with exceptionally high inference demands, potentially enabling so far uncomputable tasks in scientific computing.

ACKNOWLEDGMENTS

We thank Giammarco Fabiani for stimulating discussions and support on the condensed matter physics use case, as well as for providing the trained networks, and Julita Corbalan for support on operating the EAR software package. We thank our colleagues at IBM Research, in particular Y. Kohda, S. Munetoh, and G. Karunaratne, for discussions on the MP-AIMC power/performance analysis and acknowledge the support from the IBM Research AI Hardware Center. This research received funding from the Interdisciplinary Research Platform (IRP) for Green Information Technology at the Faculty of Science of Radboud University that aims to promote pioneering research ideas in emerging interdisciplinary themes, the Nederlandse Organisatie voor Wetenschappelijk Onderzoek (NWO) and is part of the Shell NWO/FOM initiative “Computational sciences for Energy Research” of Shell and Chemical Sciences, Earth and Life Sciences, Physical Sciences, Stichting voor Fundamenteel Onderzoek der Materie (FOM), and Stichting voor de Technische Wetenschappen (STW). It also received funding from the European Research Council under ERC Grant Agreement No. 856538 (3D-MAGiC), the European Union’s Horizon Research and Innovation programme under Grant Agreement No. 101046878 (HYBRAIN), and the Swiss State Secretariat for Education, Research and Innovation (SERI) under Contract No. 22.0002. We gratefully acknowledge the computer resources at Artemisa, funded by the European Union ERDF and Comunitat Valenciana as well as the technical support provided by the Instituto de Fisica Corpuscular, IFIC (CSIC-UV). R. Ruiz de Austri also acknowledges the Ministerio de Ciencia e Innovación (Grant No. PID2020-113644GB-I00).

AUTHOR DECLARATIONS

Conflict of Interest

The authors have no conflicts to disclose.

Author Contributions

Dominique J. Kösters: Data curation (lead); Formal analysis (lead); Investigation (lead); Project administration (supporting); Software (lead); Visualization (lead); Writing – original draft (lead); Writing – review & editing (equal). **Bryan A. Kortman**: Data curation (equal); Formal analysis (equal); Investigation (equal); Software (equal); Visualization (equal); Writing – original draft (equal);

Writing – review & editing (equal). **Irem Boybat**: Methodology (equal); Supervision (equal); Validation (equal); Writing – review & editing (equal). **Elena Ferro**: Formal analysis (equal); Investigation (equal); Validation (equal); Writing – review & editing (equal). **Sagar Dolas**: Conceptualization (equal); Funding acquisition (equal); Methodology (equal); Resources (equal); Writing – review & editing (equal). **Roberto Ruiz de Austri**: Data curation (equal); Funding acquisition (equal); Software (equal); Writing – review & editing (equal). **Johan Kwisthout**: Conceptualization (equal); Funding acquisition (equal); Writing – review & editing (equal). **Hans Hilgenkamp**: Writing – review & editing (equal). **Theo Rasing**: Conceptualization (equal); Funding acquisition (equal); Writing – review & editing (equal). **Heike Riel**: Conceptualization (equal); Funding acquisition (equal); Resources (equal); Writing – review & editing (equal). **Abu Sebastian**: Funding acquisition (equal); Methodology (equal); Supervision (equal); Writing – review & editing (equal). **Sascha Caron**: Conceptualization (equal); Funding acquisition (equal); Methodology (equal); Supervision (equal); Writing – review & editing (equal). **Johan H. Mentink**: Conceptualization (lead); Funding acquisition (lead); Methodology (lead); Project administration (lead); Resources (equal); Supervision (lead); Visualization (equal); Writing – original draft (lead); Writing – review & editing (lead).

DATA AVAILABILITY

The data that support the findings of this study are openly available in github at https://github.com/dkosters/EME.[34]

REFERENCES

[1] A. Mehonic and A. J. Kenyon, "Brain-inspired computing needs a master plan," Nature **604**, 255–260 (2022).

[2] K. T. Butler, D. W. Davies, H. Cartwright, O. Isayev, and A. Walsh, "Machine learning for molecular and materials science," Nature **559**, 547–555 (2018).

[3] G. Carleo, I. Cirac, K. Cranmer, L. Daudet, M. Schuld, N. Tishby, L. Vogt-Maranto, and L. Zdeborová, "Machine learning and the physical sciences," Rev. Mod. Phys. **91**, 045002 (2019).

[4] E. Bedolla, L. C. Padierna, and R. Castañeda-Priego, "Machine learning for condensed matter physics," J. Phys.: Condens. Matter **33**, 053001 (2020).

[5] A. Radovic, M. Williams, D. Rousseau, M. Kagan, D. Bonacorsi, A. Himmel, A. Aurisano, K. Terao, and T. Wongjirad, "Machine learning at the energy and intensity frontiers of particle physics," Nature **560**, 41–48 (2018).

[6] M. Feickert and B. Nachman, "A living review of machine learning for particle physics," arXiv:2102.02770 (2021).

[7] G. Karagiorgi, G. Kasieczka, S. Kravitz, B. Nachman, and D. Shih, "Machine learning in the search for new fundamental physics," Nat. Rev. Phys. **4**, 399–412 (2022).

[8] E. Govorkova, E. Puljak, T. Aarrestad, M. Pierini, K. A. Woźniak, and J. Ngadiuba, "LHC physics dataset for unsupervised New Physics detection at 40 MHz," Sci. Data **9**, 118 (2022).

[9] M. Davies, A. Wild, G. Orchard, Y. Sandamirskaya, G. A. F. Guerra, P. Joshi, P. Plank, and S. R. Risbud, "Advancing neuromorphic computing with Loihi: A survey of results and outlook," Proc. IEEE **109**, 911–934 (2021).

[10] C. D. Schuman, S. R. Kulkarni, M. Parsa, J. P. Mitchell, and B. Kay, "Opportunities for neuromorphic computing algorithms and applications," Nat. Comput. Sci. **2**, 10–19 (2022).

[11] J. B. Aimone, O. D. Parekh, and W. M. Severa, "Neural computing for scientific computing applications," in Neuromorphic Computing 2017, Knoxville, TN, July 2017.

[12] G. A. Fonseca Guerra and S. B. Furber, "Using stochastic spiking neural networks on spinnaker to solve constraint satisfaction problems," Front. Neurosci. **11**, 714 (2017).

[13] S. Czischek, A. Baumbach, S. Billaudelle, B. Cramer, L. Kades, J. M. Pawlowski, M. K. Oberthaler, J. Schemmel, M. A. Petrovici, T. Gasenzer, and M. Gärttner, "Spiking neuromorphic chip learns entangled quantum states," SciPost Phys. **12**, 39 (2022).

[14] A. Sebastian, M. Le Gallo, R. Khaddam-Aljameh, and E. Eleftheriou, "Memory devices and applications for in-memory computing," Nat. Nanotechnol. **15**, 529–544 (2020).

[15] P. Yao, H. Wu, B. Gao, J. Tang, Q. Zhang, W. Zhang, J. J. Yang, and H. Qian, "Fully hardware-implemented memristor convolutional neural network," Nature **577**, 641–646 (2020).

[16] L. Fick, S. Skrzyniarz, M. Parikh, M. B. Henry, and D. Fick, "Analog matrix processor for edge AI real-time video analytics," in *IEEE International Solid-State Circuits Conference (ISSCC)* (IEEE, 2022), Vol. 65, pp. 260–262.

[17] G. Carleo and M. Troyer, "Solving the quantum many-body problem with artificial neural networks," Science **355**, 602–606 (2017).

[18] G. Fabiani and J. H. Mentink, "Investigating ultrafast quantum magnetism with machine learning," SciPost Phys. **7**, 4 (2019).

[19] T. Aarrestad, M. van Beekveld, M. Bona, A. Boveia, S. Caron, J. Davies, A. de Simone, C. Doglioni, J. M. Duarte, A. Farbin, H. Gupta, L. Hendriks, L. Heinrich, J. Howarth, P. Jawahar, A. Jueid, J. Lastow, A. Leinweber, J. Mamuzic, E. Merényi, A. Morandini, P. Moskvitina, C. Nellist, J. Ngadiuba, B. Ostdiek, M. Pierini, B. Ravina, R. Ruiz de Austri, S. Sekmen, M. Touranakou, M. Vaškeviciute, R. Vilalta, J.-R. Vlimant, R. Verheyen, M. White, E. Wulff, E. Wallin, K. A. Wozniak, and Z. Zhang, "The dark machines anomaly score challenge: Benchmark data and model independent event classification for the large Hadron collider," SciPost Phys. **12**, 43 (2022).

[20] S. Caron, L. Hendriks, and R. Verheyen, "Rare and different: Anomaly scores from a combination of likelihood and out-of-distribution models to detect new physics at the LHC," SciPost Phys. **12**, 77 (2022).

[21] Y. Nomura and M. Imada, "Dirac-type nodal spin liquid revealed by refined quantum many-body solver using neural-network wave function, correlation ratio, and level spectroscopy," Phys. Rev. X **11**, 031034 (2021).

[22] G. Fabiani, M. D. Bouman, and J. H. Mentink, "Supermagnonic propagation in two-dimensional antiferromagnets," Phys. Rev. Lett. **127**, 097202 (2021).

[23] S. Sorella, "Green function Monte Carlo with stochastic reconfiguration," Phys. Rev. Lett. **80**, 4558–4561 (1998).

[24] K. Ido, T. Ohgoe, and M. Imada, "Time-dependent many-variable variational Monte Carlo method for nonequilibrium strongly correlated electron systems," Phys. Rev. B **92**, 245106 (2015).

[25] C. Bierlich, S. Chakraborty, N. Desai, L. Gellersen, I. Helenius, P. Ilten, L. Lönnblad, S. Mrenna, S. Prestel, C. T. Preuss, T. Sjöstrand, P. Skands, M. Utheim, and R. Verheyen, "A comprehensive guide to the physics and usage of PYTHIA 8.3," arXiv:2203.11601 (2022).

[26] S. Ovyn, X. Rouby, and V. Lemaitre, "DELPHES, a framework for fast simulation of a generic collider experiment," arXiv:0903.2225 (2009).

[27] J. Corbalan, "Energy aware runtime (EAR) documentation," https://www.bsc.es/research-and-development/software-and-apps/software-list/ear-energy-management-framework-hpc (2017).

[28] R. Khaddam-Aljameh, M. Stanisavljevic, J. Fornt Mas, G. Karunaratne, M. Brändli, F. Liu, A. Singh, S. M. Müller, U. Egger, A. Petropoulos, T. Antonakopoulos, K. Brew, S. Choi, I. Ok, F. L. Lie, N. Saulnier, V. Chan, I. Ahsan, V. Narayanan, S. R. Nandakumar, M. Le Gallo, P. A. Francese, A. Sebastian, and E. Eleftheriou, "HERMES-core—A 1.59-TOPS/mm^2 PCM on 14-nm CMOS in-memory compute core using 300-ps/LSB linearized CCO-based ADCs," IEEE J. Solid-State Circuitss **57**, 1027–1038 (2022).

[29] P. Narayanan, S. Ambrogio, A. Okazaki, K. Hosokawa, H. Tsai, A. Nomura, T. Yasuda, C. Mackin, S. Lewis, A. Friz *et al.*, "Fully on-chip MAC at 14 nm enabled

by accurate row-wise programming of PCM-based weights and parallel vector-transport in duration-format," in *2021 Symposium on VLSI Technology* (IEEE, 2021), pp. 1–2.

[30] M. Le Gallo, A. Sebastian, R. Mathis, M. Manica, H. Giefers, T. Tuma, C. Bekas, A. Curioni, and E. Eleftheriou, "Mixed-precision in-memory computing," Nat. Electron. **1**, 246–253 (2018).

[31] I. Giannopoulos, A. Sebastian, M. Le Gallo, V. P. Jonnalagadda, M. Sousa, M. Boon, and E. Eleftheriou, "8-bit precision in-memory multiplication with projected phase-change memory," in *2018 IEEE International Electron Devices Meeting (IEDM)* (IEEE, 2018), p. 27.

[32] M. Le Gallo, S. R. Nandakumar, L. Ciric, I. Boybat, R. Khaddam-Aljameh, C. Mackin, and A. Sebastian, "Precision of bit slicing with in-memory computing based on analog phase-change memory crossbars," Neuromorphic Comput. Eng. **2**, 014009 (2022).

[33] V. Joshi, M. Le Gallo, S. Haefeli, I. Boybat, S. R. Nandakumar, C. Piveteau, M. Dazzi, B. Rajendran, A. Sebastian, and E. Eleftheriou, "Accurate deep neural network inference using computational phase-change memory," Nat. Commun. **11**, 2473 (2020).

[34] D. J. Kosters, B. A. Kortman, R. Ruiz de Austri, and G. Fabiani (2022). "EME," Github. https://github.com/dkosters/EME.