arXiv:2405.03605v1 [cs.NE] 6 May 2024

Trackable Island-model Genetic Algorithms at Wafer Scale

Matthew Andres Moreno
Connor Yang
University of Michigan
Ann Arbor, Michigan, USA

Emily Dolson
Michigan State University
East Lansing, Michigan, USA

Luis Zaman
University of Michigan
Ann Arbor, Michigan, USA

ABSTRACT

Emerging ML/AI hardware accelerators, like the 850,000 processor Cerebras Wafer-Scale Engine (WSE), hold great promise to scale up the capabilities of evolutionary computation. However, challenges remain in maintaining visibility into underlying evolutionary processes while efficiently utilizing these platforms' large processor counts. Here, we focus on the problem of extracting phylogenetic information from digital evolution on the WSE platform. We present a tracking-enabled asynchronous island-based genetic algorithm (GA) framework for WSE hardware. Emulated and on-hardware GA benchmarks with a simple tracking-enabled agent model clock upwards of 1 million generations a minute for population sizes reaching 16 million. This pace enables quadrillions of evaluations a day. We validate phylogenetic reconstructions from these trials and demonstrate their suitability for inference of underlying evolutionary conditions. In particular, we demonstrate extraction of clear phylometric signals that differentiate wafer-scale runs with adaptive dynamics enabled versus disabled. Together, these benchmark and validation trials reflect strong potential for highly scalable evolutionary computation that is both efficient and observable. Kernel code implementing the island-model GA supports drop-in customization to support any fixed-length genome content and fitness criteria, allowing it to be leveraged to advance research interests across the community.

CCS CONCEPTS

• **Computing methodologies** → **Genetic algorithms**; **Artificial life**; *Genetic programming*; **Massively parallel and high-performance simulations**; *Agent / discrete models.*

KEYWORDS

island-model genetic algorithm, phylogenetics, wafer-scale computing, evolutionary computation, high-performance computing, Cerebras Wafer-Scale Engine, agent-based modeling, phylogenetic tracking, evolutionary computation

ACM Reference Format:
Matthew Andres Moreno, Connor Yang, Emily Dolson, and Luis Zaman. 2024. Trackable Island-model Genetic Algorithms at Wafer Scale. In *Genetic and Evolutionary Computation Conference (GECCO '24 Companion), July 14–18, 2024, Melbourne, VIC, Australia.* ACM, New York, NY, USA, 2 pages. https://doi.org/10.1145/3638530.3664090

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for third-party components of this work must be honored. For all other uses, contact the owner/author(s).
GECCO '24 Companion, July 14–18, 2024, Melbourne, VIC, Australia
© 2024 Copyright held by the owner/author(s).
ACM ISBN 979-8-4007-0495-6/24/07
https://doi.org/10.1145/3638530.3664090

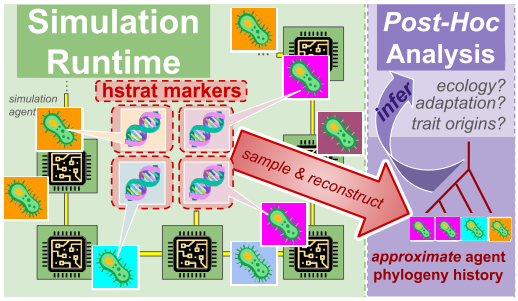


Figure 1: Strategy for trackable distributed evolution simulation. Hstrat markers attached to agents support post hoc estimation of relatedness between lineages, enabling approximate phylogenetic reconstruction.

1 INTRODUCTION

A major upshot of the deep learning race is the emergence of spectacularly capable next-generation compute accelerators [1, 2, 4, 11]. Although tailored expressly to deep learning workloads, these hardware platforms represent an exceptional opportunity to springboard the capabilities of evolutionary computation. The emerging class of fabric-based accelerators, led by the 850,000 core Cerebras CS-2 Wafer Scale Engine (WSE) [3], holds particular promise in this regard. This architecture interfaces multitudinous processor elements (PEs) in a physical lattice, with PEs executing independently and interacting locally through a network-like interface.

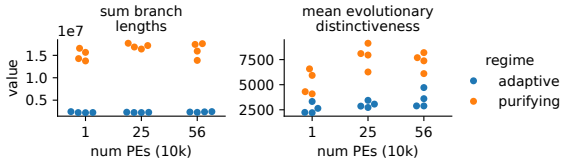
Our work here explores how such hardware might be recruited for large-scale, telemetry-enabled digital evolution, demonstrating a GA implementation tailored to the dataflow-oriented computing model of the CS-2 platform. We apply hereditary stratigraphy algorithms to enable *post hoc* reconstruction of phylogenetic history from special fitness-neutral annotations on genomes (Figure 1) [5] .

2 METHODS

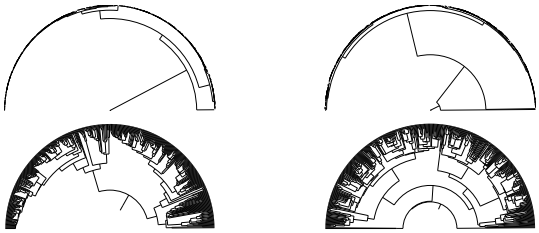
We apply an island-model genetic algorithm, common in applications of parallel and distributed computing to evolutionary computation, to instantiate a wafer-scale evolutionary process spanning the Wafer-Scale Engine's PEs. Under this model, PEs each host independent populations and exchange genomes with neighbors.

Our implementation unfolds according to a generational update cycle, with migration handled first. Each PE maintains independent immigration buffers and emigration buffers dedicated to each cardinal neighbor. Asynchronous operations are registered to on-completion callbacks that set a per-buffer "send-" or "receive-complete" flag variable. The main population buffer held 32 genomes.

Subsequently, the main update loop tests all completion flags. For each immigration flag that is set, buffered genomes are copied into the main population buffer, replacing randomly chosen population members. Likewise, for each emigration flag set, corresponding send buffers are re-populated with randomly sampled genomes from the



(a) adaptive vs. purifying phylometric structure



(b) adaptive regime **(c) purifying regime**

Figure 2: On-hardware Trial. **Results from phylogenetic reconstruction of 1 million generation on-hardware simulations. Panel 2a compares phylometric readings from purifying-only and adaptation-enabled configurations, 4 replicates each. Panels 2b and 2c juxtapose example 750×750 PE simulation phylogenies under each configuration regime. Phylometrics were calculated from reconstructions with 10k sampled end-state agents. For legibility, phylogeny visualizations were further subsampled to 1k end-state agents. Top phylogenies are linear-scaled. Bottom phylogenies are log-scaled with ultrametric correction to better show topology.**

main population buffer. Corresponding flags are then reset and new async requests are initiated.

The remainder of the main update loop handles evolutionary operations within the scope of the executing PE. Each genome within the population is evaluated to produce a scalar fitness value. After evaluation, tournament selection is applied. Each slot in the next generation is populated with the highest-fitness genome among $n=5$ randomly sampled individuals, with ties broken randomly.

Finally, a mutational operator is applied across all genomes in the next population. At this point, hereditary stratigraphy annotations are updated to reflect an elapsed generation. The next generation cycle then launches, unless a generation count halting condition is met.

Kernel code for the island-model GA is open source at `github.com/mmore500/wse-sketches/tree/v0.2.2`, and can be harnessed for any fixed-length genome model and evaluation criteria by drop-in replacement of one source file containing three functions (initialize, mutate, and evaluate). We hope to see this work contribute to scale-up in use cases across the research community. We used Cerebras SDK v1.0.0 [10], which allows code to be developed and tested regardless of access to Cerebras hardware. Hereditary stratigraphy utilities were used from the *hstrat* Python package [6]. For more details, see our upcoming publication on this work [9].

3 RESULTS AND DISCUSSION

For on-hardware proof-of-concept experiments, we used a 128-bit, tracking-enabled genome layout, with the full first 32 bits containing a floating point fitness value. We defined two treatments: *purifying-only*, where 33% of agent replication events decreased fitness by a normally-distributed amount, and *adaption-enabled*, which added beneficial mutations that increased fitness by a normally-distributed amount, occurring with 0.3% probability. These beneficial mutations introduced the possibility for selective sweeps. As before, we used tournament size 5 for both treatments. We performed four on-hardware replicates of each treatment instantiated on 10k (100×100), 250k (500×500) and 562.5k (750×750) PE arrays. We halted each PE after it elapsed 1 million generations.

Across eight on-device, tracking-enabled trials of 1 million generations, we measured a mean simulation rate of 17,688 generations per second for 562,500 PEs (750×750 rectangle) with run times slightly below one minute. Trials with 1,600 PEs (40×40) performed similarly, completing 17,734 generations per second.

Multiplied out to a full day, 17 generations per second turnover would elapse around 1.5 billion generations. With 32 individuals hosted per each of 850,000 PEs, the net population size would sit around 27 million at full CS-2 wafer scale. (Note, though, that available on-chip memory could support thousands of our very simple agents per PE, raising the potential for a net population size on the order of a billion agents.) A naive extrapolation estimates on the order of a quadrillion agent replications could be achieved hourly at full wafer scale for such a very simple model. We look forward to more benchmarking and scaling experiments in future work.

Upon completion, we sampled one genome from each PE. Then, we performed an agglomerative trie-based reconstruction from subsamples of 10k end-state genomes [8]. Figure 2 compares phylogenies generated under the purifying-only and adaption-enabled treatments. As expected [7], purifying-only treatment phylogenies consistently exhibited greater sum branch length and mean evolutionary distinctiveness, with the effect on branch length particularly strong. These structural effects are apparent in example phylogeny trees from 562.5k PE trials (Figures 2b and 2c). Successful differentiation between treatments is a highly promising outcome. This result not only supports the correctness of our methods and implementation, but also confirms the capability of reconstruction-based analyses to meaningfully describe very large-scale evolving populations.

ACKNOWLEDGMENTS

Supported in part by the Schmidt AI in Sci Fellowship. Thank you also to M. Jacquelin, U. Mody, and L. Wilson at Cerebras Systems.

REFERENCES

[1] Emani et al. 2021. Accelerating scientific applications with sambanova reconfigurable dataflow architecture. *Comp. in Science & Engineering* 23, 2 (2021), 114–119.
[2] Jia et al. 2019. Dissecting the graphcore ipu architecture via microbenchmarking. *arXiv preprint arXiv:1912.03413* (2019).
[3] Lauterbach. 2021. The Path to Successful Wafer-Scale Integration: The Cerebras Story. *IEEE Micro* 41, 6 (2021), 52–57. https://doi.org/10.1109/mm.2021.3112025
[4] Medina and Dagan. 2020. Habana labs purpose-built ai inference and training processor architectures. *IEEE Micro* 40, 2 (2020), 17–24.
[5] Moreno, Dolson, and Ofria. 2022. Hereditary Stratigraphy: Genome Annotations to Enable Phylogenetic Inference over Distributed Populations *(The 2022 Conference on Artificial Life, 80)*. MIT Press, 64. https://doi.org/10.1162/isal_a_00550
[6] Moreno, Dolson, and Ofria. 2022. hstrat: a Python Package for phylogenetic inference on distributed digital evolution populations. *Journal of Open Source Software* 7, 80 (2022), 4866. https://doi.org/10.21105/joss.04866
[7] Moreno, Dolson, and Rodriguez Papa. 2023. Toward Phylogenetic Inference of Evolutionary Dynamics at Scale. In *Proceedings of the 2023 Artificial Life Conference*. 79. https://doi.org/10.1162/isal_a_00694
[8] Moreno, Papa, and Dolson. 2024. Analysis of Phylogeny Tracking Algorithms for Serial and Multiprocess Applications. https://doi.org/10.48550/arXiv.2403.00246
[9] Moreno, Yang, Dolson, and Zaman. 2024. Trackable Agent-based Evolution Models at Wafer Scale. https://arxiv.org/abs/2404.10861
[10] Selig. 2022. The cerebras software development kit: A technical overview. (2022).
[11] Zhang et al. 2016. Cambricon-X: An accelerator for sparse neural networks. In *International Symposium on Microarchitecture*. Ieee, 1–12.