Next in Reversible Computing: Breaking the Memory-Computation Asymmetry

Kalyan Perumalla
Oak Ridge National Laboratory[1]
perumallaks@ornl.gov

In the author's opinion, classical computational speed is the biggest nemesis of reversible computing. The high speed of classical irreversible computation is indomitable, albeit at the expense of energy dissipated as heat. In practical applications, part of the high speed comes from memory support. Classical memory techniques such as caching, memoization, and simple reset semantics of memory pervade classical computing, making it more difficult for reversible computing to become competitive with classical irreversible computing.

Theoretical research in reversible computing has enjoyed treatment at the level of abstractions such as Turing Machines with respect to computability, space complexity, time complexity, and completeness. However, practical research in reversible computing has predominantly been tackled at the processor level. To enable the next leaps in the practical realization of reversible computing, a renewed emphasis is useful on uncovering and resolving memory-related aspects in reversibility.

A reversibility-driven approach to the relation between *memory* and *computation* would lead to at least two important outcomes: (1) It would result in a more fundamental understanding of their nature to help answer questions such as: Is memory a concept fundamentally distinct from the concept of computation? Or, are memory and computation complementary to each other? Or, are they essentially transformations of each other? (2) It would also help clarify the role that memory plays along key computational dimensions such as speed and energy usage, making it possible for reversible computing to be a competitive, practical alternative to classical irreversible computing.

Models such as Pebble Games have been very effective in uncovering the fundamental tradeoffs between space (memory) and time (computation) in reversible execution, exposing the corresponding theoretical complexities underlying all erasure-free reversible computation. Partial erasure models have gone further and provided a good approach to exploring the region between purely reversible computation and classical irreversible computation. However, these treatments have largely remained theoretical in nature. A more comprehensive and practical treatment of memory remains to be performed in the broad field of reversible computing, visiting and answering memory-related unknowns. Some of the answers could have profound impact on classical memory technologies. They have the potential to touch fundamental assumptions and views about the "physics of memory" when considered in conjunction with, or in isolation from, computation.

Current knowhow in reversible computing offers abstractions such as swap operations in order to preserve reversibility of stored (non-volatile) memory, including file systems. This initial approach needs to be expanded to cover other dimensions such as speed and energy costs. Just as classical (irreversible) computation has been revisited and relaxed to cover speed and energy costs, memory also merits a new, analogous treatment. For example, how would the dimension of volatility intersect reversibility? What are some of the physical limits that will be encountered, and how do they intersect the

established concepts in making computation reversible? What are the thermodynamic arguments (such as asymptotic isentropy) that come into play in such settings? How do they translate to physical processes and device technologies? How are space and time complexity measures affected when practical memory device models are accounted for? Is there a grand, unified reversible sequential circuit theory that would be possible to develop?

A practical context in which the memory-computation asymmetry manifests is in ensembles of computation. Consider a tree of computation (such as ensembles of "cloned" simulations) in which child nodes are small perturbations of their parent node. Let the tree be of k levels, with m children per node, and f be the fraction of change between parent and child. In such ensembles, the factor of reduction in aggregate memory compared to replicated runs is $1/(m^{1-k}+f)$. However, the factor of reduction in aggregate computation compared to replicated runs is $1/(m^{1-k}+f/k)$, which is more than the gain in memory. This is directly based on classical memory semantics where it is possible to reinitialize the memory with zero cost, representing the memory-computation asymmetry. However, reversible execution restores symmetry between memory and computation, correctly reinstating the memory state restoration cost in the aggregate memory cost. The symmetry is obtained through a normalization by combining the computational cost with memory restoration cost between ensemble runs.

Other practical aspects include consideration of the necessarily varying speeds of different types of memory technologies and their associated manufacturing economic costs, which results in the deep memory hierarchies of modern irreversible computing. The gamut of associated issues, such as compilation techniques and optimization techniques will need to be reconciled with reversible execution semantics for the computational portions. Locality of references, working set definitions, and spatio-temporal caching are all memory-related challenges that will require adequate resolution for future success of reversible computing. Circuit layout algorithms for reversibility at the hardware levels will have to carefully account for, and resolve, this memory-computation asymmetry.

One of the potential approaches to relaxing the memory interface is the creation of an "anti-memoization" infrastructure. Another approach is that of a unified memory-compute infrastructure that includes miniature versions of the compute-copy-uncompute trick of reversible computation, realized in a recursively hierarchical structure of a hybrid memory-computation hardware abstraction.

Overall, one of the important inflection points that is key to the advancement of reversible computing appears to lie in breaking the current asymmetry between memory and computation.

References

[1] Rolf Landauer, "Irreversibility and Heat Generation in the Computing Process," IBM Journal of Research and Development, Vol. 5(3), 1961
[1] Charles H. Bennett, "Logical reversibility of computation," IBM Journal of Research and Development Vol. 17, 1973
[3] Kalyan S. Perumalla, "Introduction to Reversible Computing," (book), Taylor and Francis, Chapman and Hall/CRC, 1st Edition, ISBN 978-1439873403, 2014
[4] Srikanth B. Yoginath and Kalyan Perumalla, "Scalable cloning on Large-scale GPU platforms with Application to Time-stepped Simulations on Grids," ACM Transactions on Modeling and Computer Simulation, Vol. 28(1), 2018
[4] Michael P. Frank, "Throwing Computing into Reverse," IEEE Spectrum, Vol. 54(9), 2017
[5] Srikanth B. Yoginath, Maksud Alam and Kalyan S. Perumalla, "Energy Conservation through Cloned Execution of Simulations," in Proceedings of the Winter Simulation Conference, 2019