A Candidate Approach for Optical In-Network Computation

Joe Touch
USC/ISI
Marina del Rey, CA 90292-6695 USA
touch@isi.edu

Yinwen Cao, Morteza Ziyadi, Ahmed Almaiman, Amirhossein Mohajerin Ariaei, Alan E. Willner
Department of Electrical Engineering
USC
Los Angeles, CA

***Abstract*—Issues in developing support for in-network computation on optical signals are presented. Key challenges and concerns with past attempts are addressed, including difficulties supporting high-level computational models. A candidate approach using frequency mixing of M-PSK signals for both computation and transmission is presented.**

Keywords—optical computation, nonlinear optics, digital optics

I. Introduction

The search for a viable approach to optical computing has spanned the past seven decades and there has been recent interest in exploring the potential for optical computing for a variety of uses [1][2]. Here we explore the use of optical computing to support in-network computation because long-distance, high-speed communication is necessarily optical. In the process, a number of key design properties are determined and the issues in a practical way forward are developed.

Computing is already very efficient in the electronic domain, so the first question is "why optics?" The answer lies in long-distance, high-speed transmission – signals over 10 Gbps are difficult to transmit over 10 meters; this relationship persists as a product (i.e., 1 Gbps over 100 meters, 100 Mbps over 1 km, etc.). So we begin with the observation that "going fast" means going optical, which begs the next question: is there a need to compute on data while it is in transit and if so, should we compute in optics? The answer to both are "yes."

Data in-transit needs to be dynamically redirected, to be routed according to context either in-band or out-of-band. That data can also exceed the capacity of end system storage, e.g., when aggregating from sampling sources, and it is useful to winnow that flow down to a manageable size, either for big data filtering, content aggregation, or compression. Finally, data in transit benefits from encryption for two reasons: site-to-site content and flow activity protection as well as leverage cross-flow entropy for stronger encryption.

II. The Challenges of Optical Computation

As part of our Optical Turing Machine (OTM) project, we are exploring the challenges of developing digital optical computation to support these in-transit operations [3].

A. Optics Issues

Merely replacing electronics with optics remains a significant challenge. Photons do not interact in ordinary environments; they interact only indirectly through matter. These interactions typically require either out-of-band electronic control (e.g., by inducing a voltage or current on the material) or high-power optical signals (e.g., for frequency mixing). As a result, we do not assume that a shift to optical computing will result in energy reduction.

Optical frequencies are 100x larger than commercial IC processes (1500 nm vs. 14 nm), so optics will require 10,000x more area for an equivalent device. Electronic switching is capable of over 600 Gbps using SiGe devices, where optical switching is 15x slower (40 GHz for LiNO3). Only frequency mixing provides the needed speed to overcome these limits.

B. Computing issues

When developing any type of computing, it is important to understand the target computational model. It is tempting to focus on computational complexity, which compares the time to compute a solution, but computing models define what types of problems can be solved at all. They escalate from basic combinatorial logic (CL), to finite state machines (FSM, recirculating one state), to pushdown automata (PDA, "infinite" state, but accessing only the most recent state), to Turing Machines (TM, infinite state with arbitrary access).

There are two very important differences in considering these models for electronic and optical computation. First, all states beyond the FSM model may need to persist for an arbitrarily long time. Second, electronics can support persistent state using combinatorial logic with infinite feedback. Neither of these apply to optics because light cannot be "stopped" and because feedback incurs noise that is hard to dampen.

C. Past approaches

We characterize past approaches to optical computing as typically either "aluminum feathers" or "Field of Dreams"[1]. The "aluminum feather" analogy is: birds fly, airplanes fly; birds have feathers, so airplanes must need feathers. Optical transistors are similar to aluminum feathers – they are sought largely by analogy: because electronics computes with switching, so must optics [4]. However, this discounts a few key points: optics switches much more slowly than electronics, optical devices are 10,000x larger in area, and optical transmission typically uses multibit symbols rather than binary.

[1] Refers to the movie and its line: "if you build it, they will come", i.e., a "build first, apply later" design approach.

This work was partly supported by NSF INSPIRE Grant #1344221.

The "Field of Dreams" approach has similar concerns. Many devices and systems have been developed in the expectation that they might be useful for optical computation (sometime in addition to another, primary use) [5][6], but many of these approaches directly interfere with computation itself. Our own team investigated time-differential processing to reduce noise, which takes an ABCD… stream and creates a corresponding (A-B)(B-C)(C-D)… stream [7]. We later realized that computing on this transformed stream was not feasible, nor was it possible to recover the original sequence without inserting symbols that also interfered.

III. A Candidate Approach

As a consequence of these observations, in-network computation might benefit from a specific approach – one that relies primarily on frequency mixing using encodings that support long-distance transmission [3]. Of the possible alternatives, only M-PSK suffices; QAM and its variants cannot be processed without the use of either conditional mechanisms (i.e., add one way for one power, add a different way for another) or demultiplexing (i.e., "OOO" – converting the transmission format to a different computation format). The former is too complex and the latter equivalent to OEO – converting optical transmission into electronics for computation. The goal is to avoid both issues.

Processing in-transit information places a premium on serial operations. Although not novel to computing, they were largely abandoned for parallelism because electronics needs to parallelize computation to achieve high speed. Optics, notably frequency mixing, can support multi-THz signals in serial, so computing in that native format is also a goal.

Frequency mixing already supports most of the operations needed for a field – a set of symbols with two binary operators, each supporting a group. Boolean logic is one example of a field; modulus arithmetic, e.g., on M-PSK symbols, is another. Re-programmability can be supported using reconfiguration, e.g., using Mach-Zehnder lossless switching (at relatively low speeds because this need not be a von Neumann architecture, i.e., with instructions and memory in the same area). The result is more similar to systolic or data-flow architectures than the more conventional electronic register-based architectures.

IV. Remaining Challenges

A number of key challenges remain in realizing an optical computer, even when limiting it to in-network operations. The two most important are storage and regeneration, and they are very closely related. Storage is also the key to higher order computation capability, necessary for anything beyond a FSM.

In electronics, there are two different ways to achieve storage: dynamic feedback and static state. Dynamic feedback uses combinatorial logic with infinite recirculation to store state, e.g., using two NOR gates to implement an RS-flipflop. Static state stops time, either parking electrons themselves or converting to other representations that can be parked. Infinite recirculation relies on perfect symbol regeneration and stopping time requires temporally-stable representations, neither of which are currently available for optics.

In optics, lookback can be achieved with a delay line, allowing the input stream to interact with past state with limited reach. This can support either a FSM or limited-lookback TM, e.g., allowing recent past input to provide context to current input processing. The issue of indefinite state recirculation is more challenging requiring regeneration.

State regeneration allows electronic logic devices to be cascaded in arbitrary ways, either an arbitrary-depth chain or using feedback, where feedback enables using logic for state. Our team has explored regeneration of M-PSK symbols, using phase squeezing or phase-sensitive amplification (PSA) for phase and saturation amplification to restore amplitude [8]. Other challenges remain, notably the need for more complex functions such as carry generation for modulus arithmetic, which involves cascading multiple frequency mixing processes in succession, but without the opportunity to do phase restoration between the stages. We are also currently evolving existing phase restoration methods to avoid the use of either differential processing or the use of pilot signals, because both interfere with the use of frequency mixing for computation.

V. The Path Forward

In the process of developing practical digital optical computation, we are currently exploring degenerate PSA to overlay the pilot with the data signal. We are developing new "light-friendly" algorithms, i.e., serial computing with limited lookback and limited recirculation of state, notably to support optical encryption and integrity protection.

Finally, we are exploring the use of hybrid ICs for implementation. Our recent results confirm significant challenges for benchtop experiments that rely on coherent frequency mixing. Using separate IC technologies poses similar challenges when aligning inputs and the variety of pumps and processed signals. Only a hybrid approach can support pump generation, frequency mixing, switching (for programmability), and passive filters and couplers on a single substrate to avoid complex dynamic phase realignment.

References

[1] P. Ambs, "Optical Computing: A 60-Year Adventure," *Adv. In Optical Technologies*, 2010.

[2] OSA Optical Computing Incubator meeting, Wash. DC, Dec. 2015. http://www.osa.org/en-us/opticalcomputing/

[3] J. Touch, A. Willner, "Native Digital Processing for Optical Networking," Proc. IEEE Third Int'l. Conference on Future Generation Communication Technologies (FGCT), Aug. 2014.

[4] D. Miller, "Are optical transistors the next logical step?" *Nature Photonics*, V4, N3-5, 2010.

[5] D. Jackson, "Photonic processors: a systems approach," *Applied Optics*, V33, N23, Aug. 1994.

[6] D. Feitelson, *Optical Computing: A Survey for Computer Scientists*, MIT Press, 1992.

[7] J. Touch, A. Mohajerin-Ariaei, M. Chitgarha, M. Ziyadi, S. Khaleghi, Y. Akasaka, J.-Y. Yang, M. Sekiya, "The Impact of Errors on Differential Optical Processing," USC/ISI Tech. Report ISI-TR-690, Mar. 2014.

[8] J.-Y. Yang, Y. Akasaka, M. Ziyadi, A. Mohajerin-Ariaei, Y. Cao, A. Almaiman, I. Kim, J. Touch, A. Willner, M. Sekiya, "PSA and PSA-Based Optical Regeneration for Extending the Reach of Spectrally Efficient Advanced Modulation Formats," IEEE Summer Topicals 2015.