The Promise of Optical Free-Space Interconnections for Concurrent Memory Access

Dror G. Feitelson*
IBM T. J. Watson Research Center
P. O. Box 218
Yorktown Heights, NY 10598

Larry Rudolph
Department of Computer Science
The Hebrew University of Jerusalem
91904 Jerusalem, Israel

Abstract

The concurrent-read concurrent-write (CRCW) model of shared memory for parallel computation is considered unimplementable in practice, due to limitations of electronic technology. But optical technology opens new possibilities that may overcome these limitations. An optical CRCW memory can be designed, in principle, based on the following ideas. First, optical free-space interconnections eliminate the need for shared switches that suffer from contention. Second, practically unlimited fan-in on detectors allows for concurrent writing without serialization. And third, retroreflective optical elements that reverse the propagation of beams of light implement concurrent reads without address decoding and sequential service. It should be noted, however, that the required technology is still immature, and much work is needed before a practical design is reached.

Index terms: Concurrent-access shared memory, Optical free-space interconnections, Retroreflective devices.

*Research conducted while at Hebrew University.

1 Introduction

The simplest and most widely used model of parallel computation is the parallel random access machine (PRAM), where a number of processors share the use of a random access memory [17, 35]. Data may be passed from one processor to another by writing it to the shared memory and reading it from there. As all the processors have access to all the memory cells, situations in which a number of processors access the same cell simultaneously may arise. This feature is often useful in the design of parallel algorithms; if it is supported, the model is dubbed concurrent-read concurrent-write, or CRCW PRAM [38]. The implementation of concurrent access is considered difficult, however, so algorithm designers tend to make an effort not to use simultaneous access to the same cell (e.g. [40, 9]). The model in which all cells are shared but not accessed simultaneously is called exclusive-read exclusive-write, or EREW PRAM [38].

The implementation of large-scale shared memory parallel computers has been an elusive goal. While small-scale systems can be built using a bus [4] or crossbar switch [42], the only approach suitable for large-scale machines seems to be multistage interconnection networks [18, 32, 33]. The main problem with multistage networks is the high latency that they introduce — every memory access must traverse a number of stages that is logarithmic in the number of processors. At each stage, the address is decoded and a switching decision is made. Contention for a communication link may further increase the latency, by forcing an access to wait or to be resubmitted [25]. Some improvement is possible if distinct references to the same address are combined in the network [24]; interestingly, it is then just as easy (or hard) to implement a CRCW model as an EREW one.

The main problems facing any implementation of concurrent-access shared memory are the following:

1. *Addressing* — the request must be routed through the network from the processor to the appropriate memory location.

2. *Arbitration* — conflicts between independent requests that need the same link or memory module must be resolved. This includes the definition of appropriate semantics for a concurrent write operation.

3. *Selective multicast* — a concurrent read operation should result in a multicast of the requested data to the requesting processors. Apart from generating this multicast, the problems of addressing and arbitration occur on the way back as well.

The purpose of this paper is to point out that these problems can, in principle, be solved by using optical technology, leading to the design of a CRCW memory with low latency and no contention. Contention is avoided by adhering to a guideline of keeping distinct accesses totally independent, meaning that they do not require exclusive use of any shared resources. This is possible with free-space optical interconnections, but not with electronics. The next section discusses optical free-space interconnections, and presents an overview of how they may be used to implement concurrent memory access. Sections 3 and 4 contain detailed discussions of concurrent read and write, respectively. Section 5 presents the conclusions.

2 Free-Space Optical Interconnections

Networks based on electronic technology must guide the electrical signals in dedicated wires that are routed from the source of the signal to the desired destination, or from one switch to the next. The wires must be isolated and separated from each other to prevent unwanted interactions between the signals, such as crosstalk. A sizable percentage of the hardware (and cost) of large multiprocessor systems is therefore devoted to interconnection media (see e.g. the GF11 [3]).

The expenditure on interconnection media can be avoided if optical free-space interconnections are used. With this technology, beams of light traverse the distance from the source to the destination without any guiding medium [8, 15, section 7.6]. As photons do not interact with each other — as opposed to electrons which are subject to electromagnetic interactions — these beams of light can cross each other within the same volume without any ill effects. A certain volume is still needed in order to ensure full connectivity [16], but this volume is full of air rather than being full of wires and switches. This implies that optical free-space interconnections should have a strong economical advantage.

Once the cost associated with the network is eliminated, the motivation for saving hardware

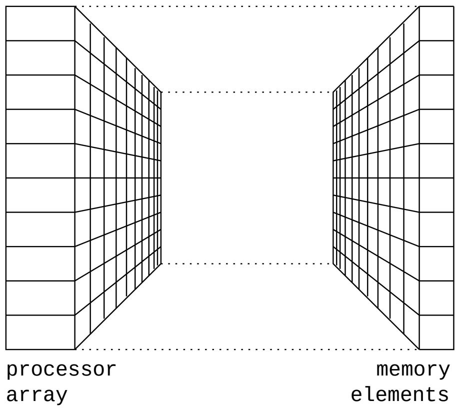

Figure 1: *A possible arrangement of processors and memory elements so that they can all "see" each other.*

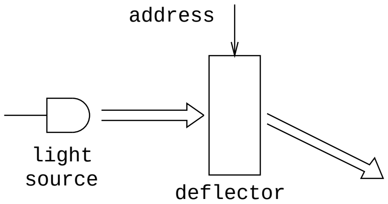

Figure 2: *The addressing mechanism for free-space interconnections is a deflector that directs the light in the desired direction.*

by sharing it among a number of communication channels disappears. Optical free-space interconnections therefore open the way for feasible implementations of large fully-connected networks, with a direct path from every source to every destination. All the intermediate switches of the network are eliminated. As a consequence, there is no danger of contention within the network. Moreover, the time needed to traverse the distance from the source to the destination is reduced, as there is no overhead for address decoding and switching along the way. Several proposals for fully connected parallel computers based on free-space optical interconnections have been made already (e.g. [34, 19, 12]).

The main requirement for the implementation of optical free-space interconnections is a line of sight between the interacting parties. In our case, these are processors and memory elements. For example, the processors can be organized in a planar array, and the memory elements in another planar array facing them, so that every processor can "see" every memory element (Fig. 1). The addressing is then done by simply aiming a beam of light directly at the addressee. To support this routing scheme, all a processor needs is a light source and a deflector. The address is used as a command to the deflector to direct the beam in a certain direction (Fig. 2). The maximal deflection angle dictates the amount of empty space that must be left between the processors and the

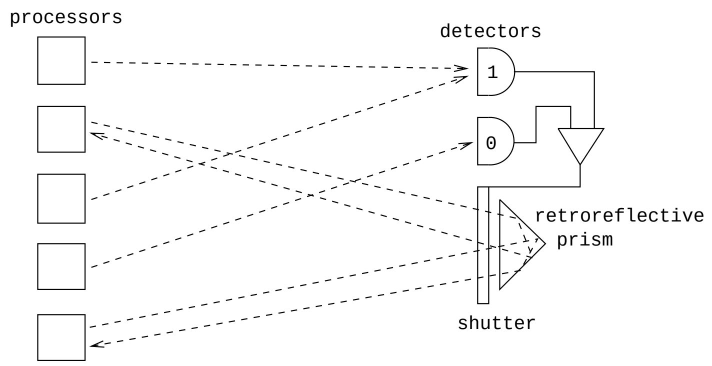

Figure 3: *Schematic representation of the concepts underlying concurrent memory access with optical free-space interconnections.*

memory [16]. Optical deflectors are readily available; the most common are based on acoustooptical phenomena [39]. Electrooptical deflectors have also been devised [22, section 14.2.2].

A largely overlooked problem with fully-connected optical free-space networks concerns collisions at the destination. If a number of processors attempt to communicate with the same destination at the same time, the data is garbled. Thus most proposals do not utilize the large fan-in possible with optics. This drawback is turned into an advantage in the case of concurrent memory access. The large fan-in possible with optics is a basic ingredient of the design, and allows the full power of a CRCW memory to be realized [7, 8].

A concurrent-access shared memory can be built based on optical free-space interconnections in the following manner (Fig. 3). Each single-bit memory element has three distinct spatial addresses that can be illuminated by processors that access it. Two of these addresses are detectors, and are used to write new data values: one is used by those processors that wish to write a "0", and the other by those writing a "1". Any number of processors can write the same or conflicting values at the same time. The outputs of the detectors are then compared to determine the stored value.

The third spatial address is used by those processors that want to read the stored value. This location is a retroreflective prism that reflects any light that hits it back towards the light source (i.e. back to the reading processor). It is guarded by a shutter that is activated by the output of the two detectors cited above. If the output of the "0" detector dominates, the shutter is opaque and no light is reflected. The processors note the absence of a reflected beam, and conclude that the stored value is “0”. If, on the other hand, the “1” detector dominates, the shutter is clear and any arriving light is reflected back to the reading processors, indicating a stored value of “1”. These ideas are elaborated below.

3 Implementation of Concurrent Read

The main problem in the implementation of a concurrent read mechanism is to send the data back the processors that requested it. In addition, individual service to the different processors should be avoided, as this would cause serialization and delays. In the proposed optical implementation, processors address memory elements by aiming a beam of light at them. The above problems are automatically solved by reversing the propagation of these beams, causing them to retrace their paths and return to the processors. This may be accomplished by so called *retroreflective* optical elements, e.g. corner reflector prisms [13]. Thus the return address is implicitly contained in the direction that the light comes from. No decoding or active routing are needed.

Concurrent reads are supported as follows. A processor that wants to read the stored value transmits a pulse of light towards the prism. The prism is protected by a switchable optical shutter, such as a Pockels cell [20, section 8.11.3] an electrooptic modulator [21, 6], or a multiple quantum well device [41]. If the shutter is clear, the pulse of light will be reversed by the prism, causing it to return to the processor that sent it. By sensing the returned pulse, the processor knows that the stored value is “1”. If the shutter is opaque, on the other hand, the pulse of light will be absorbed. By noting the absence of a returned pulse, the processor knows that the stored value is “0”. The presence of a returned pulse is determined by a threshold device on the output of the processor’s detector. This procedure is depicted in Fig. 4. It should be noted that a similar scheme was demonstrated experimentally more than ten years ago [23], albeit not in the context of parallel computers.

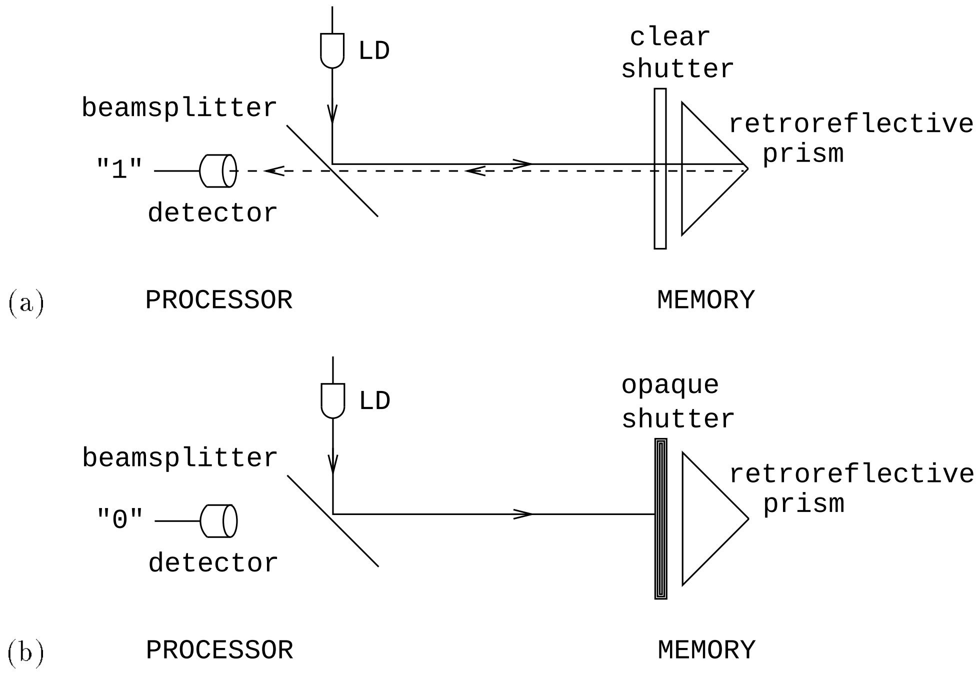

Figure 4: *The activity of a shutter is used to represent the stored value. (a) A clear shutter allows the retroreflective corner prism to reverse pulses of light falling on it, thus signaling a value of "1". (b) An opaque shutter absorbs pulses, thus signaling "0".*

Alternative Retroreflective Devices

Retroreflective prisms turn the beam around by two internal reflections; as they are passive devices, an additional active switch is needed to represent the stored value. Other devices that actively reverse the propagation of an incident beam could also be used. Switching the device on then represents a stored value of "1", while switching it off represents "0". As the reversal of the beam's propagation and its modulation are done at the same time, these devices are sometimes called *retromodulators*. In this section, we justify the choice of retroreflective prisms rather than retromodulators.

The best known active retroreflective devices are *phase conjugate mirrors* (Fig. 5) [36]. These devices can be implemented in a number of ways[1]. One of them is based on four-wave mixing,

[1] It is interesting to note that an array of retroreflective prisms also produces an effect which is very similar to phase conjugation [2].

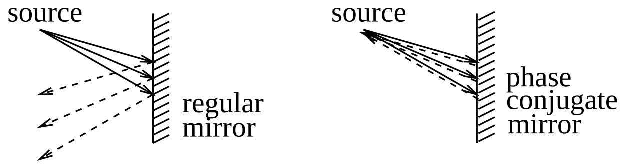

Figure 5: *A phase conjugate mirror reverses the propagation of light beams, unlike conventional mirrors that reflect light.*

where the reversed beam is generated by nonlinear interactions between the input beam and two pump beams; the required nonlinear effects occur in certain media, such as photorefractive crystals [31]. These phase conjugate mirrors reverse light beams that fall on them only if the pump beams are present [10, 43]. The presense of the pump beams, and therefore the activity of the mirror, can be used to represent the stored bit; by switching the pump beams on and off, the stored value is changed. This turns the phase conjugate mirror into a retromodulator.

The main drawback of phase conjugate mirrors is their response time. Phase conjugate mirrors based on photorefractive crystals need anywhere from milliseconds up to seconds to generate the reversed beam, which is clearly unacceptable. Once the reversed beam is generated, it can be modulated at a very high rate; this has been demonstrated in experimental settings [29, 30]. Therefore these devices may be used only when the destination — whoever generates the beam originally and aims it at the retromodulator — does not change. They are not suitable for a shared memory that may be referenced by different processors on a rapid time scale. A further disadvantage of phase conjugate mirrors is that their dynamic range might limit the number of processors that can perform a concurrent read.

The response time of photorefractive phase conjugate mirrors is reduced if light with very high intensities in used [14]. Phase conjugate mirrors based on stimulated Brillouin scattering also have short switching times, but they also require very high optical power levels. In addition, they tend to be much larger. Therefore they do not provide a realistic alternative. Another option is to use phase conjugation based on the voltage-controlled photorefractive effect. With this effect, the properties that allow the crystal to reverse the propagation of input beams are frozen, thus saving the time needed to generate the reversed beam. The modulation is done not by switching pump beams on and off, but rather by applying a voltage across the crystal [1]. Modulation rates approaching 100 KHz have been observed. The drawback of this approach is that all the possible reflection patterns have to be created in advance.

The only choice that provides retromodulation at low power and rates commensurate with processor cycle times is therefore retroreflective prisms with electrooptical shutters. The time required for the retroreflection depends on the prism size, and would be a small fraction of a nanosecond. Shutters have been demonstrated with switching times in the sub-nanosecond range [41], although they are not readily available in bulk quantities yet. A retromodulation experiment conducted more than twelve years ago achieved a switching rate of 1.4 MHz, with a view angle of 38° [23]. Better results may be expected with current technology. Moreover, the shutter might be integrated with the detectors used for concurrent writing, following the ideas used in the symmetric SEED [28].

It should be noted, however, that prisms also are not without problems. The main disadvantage of retroreflective prisms is that the effective path of the light from the processor to the memory and back is twice as long as the one-way path, which exacerbates the beam spreading due to diffraction. As phase conjugate mirrors actually reverse the propagation of the light, rather than merely reflecting it, the light is refocused on the processor when they are used[2]. Careful optical design is required to avoid crosstalk and additional reflections that cause noise.

Both retroreflective prisms and phase conjugate mirrors are also limited in the range of angles that the light comes from (called the "numerical aperture" of the device).

4 Concurrent Write and Conflict Resolution

After the addressing problem is resolved, the next problem facing a mechanism for concurrent writes is the resolution of write conflicts. In other words, there is a need to define the outcome when two or more processors attempt to write conflicting values ("0" and "1") to the same bit. The main approaches are:

[2]This is similar to the suggested use of phase conjugation for "self-targeting" in the context of SDI [36].

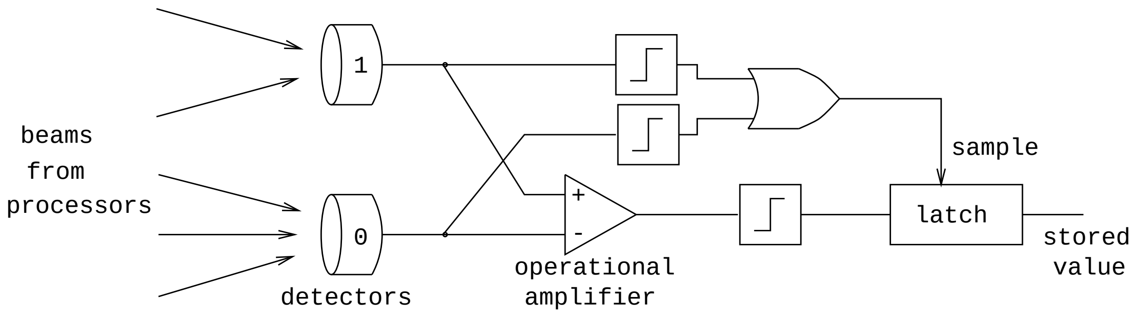

Figure 6: *The concurrent-write mechanism is based on each bit having two addresses: one for writing "1" and the other for writing "0".*

1. Assume that this does not happen, i.e. either there is no concurrent writing or else all the writers write the same value.

2. One of the values that is being written actually gets written, but the choice of which value is arbitrary.

3. The value written by the processor with the highest priority (usually taken to be the processor with the lowest ID) is written.

4. Some function of the values gets written, e.g. the minimal value ("0" in the case of binary variables), the maximal value ("1"), the majority, or the parity.

We start by presenting a system that writes either an arbitrary value or the majority value (Fig. 6; the conditions for each case and other possibilities are discussed below). Each binary memory element has two detectors, which serve as addresses that processors can aim at; one is used for writing a value of "1", and the other for writing "0". The outputs of the two detectors are compared by an operational amplifier, and the result is thresholded and latched. The latch is activated every time any detector has some output. Its output represents the stored value. The detectors and latch can be integrated with each other to form a small optical flip-flop, as was done in the symmetric SEED [28].

If all the processors write the same value, only one detector will be active, and this will dictate the stored value. If a conflict occurs, both detectors are active. In this case the written value is determined by the detector with the higher output, which represents the higher number of processors. Therefore the written value would seem to be the majority value. Due to limited dynamic range, however, this is not always true.

Conflict Resolution Schemes

Due to various quantum and statistical mechanics effects, detectors have a limited dynamic range. This means that their output should only be measured to a limited resolution; measurements with higher accuracy are meaningless. Thus if the number of processors is smaller than the number of resolvable outputs, our concurrent write mechanism actually writes the majority value. If the number of processors is larger, however, the comparison of the two detectors cannot be reliable. Therefore it is better to regard the written value as the result of an arbitrary choice.

If the number of processors is limited to the logarithm of the number of resolvable levels, conflict resolution by priorities is possible. This is done by encoding a processor's priority in the intensity of the light it emits. Assuming the processors are numbered from 0 to $p-1$, and processor 0 has the highest priority, the intensity of the beam transmitted by processor i should be $I_{max}2^{-i}$. Using this encoding, a high-priority processor can overcome the combined efforts of all the lower priority processors.

A slight change in the configuration allows for minimal-value resolution and maximal-value resolution. The change is that the output of the relevant detector ("0" for minimal value, "1" for maximal value) should have priority over the output of the other detector. This can be done by thresholding both detectors and amplifying the output of the prioritized detector before the comparison.

It is interesting to study the temporal behavior of such a memory element. Assuming that the processors are not synchronized, each may access the memory at its convenience. This means that light pulses from distinct processors may overlap to various degrees. As a consequence, the memory element might switch states during a read operation. However, this does not cause any errors, as the reading processors use a threshold device to determine whether or not the returned pulse is present. If the returned pulse is attenuated due to the switching, the processor will simply read a

"0" value, which is just as correct as a "1" value.

Multibit Memory Elements

An obvious discrepancy between the optical implementation described above and the conventional PRAM model is that the optical implementation only supports CRCW bits, whereas the PRAM calls for shared words. Concurrent reading of a whole word is easily accomplished by using k beams to address k binary memory elements in parallel. However, there is no simple solution for the concurrent writing of k bits simultaneously. The problem is that the conflict resolution is performed independently on a bit-by-bit basis, and it is therefore possible for the final vector of k bits that gets written not to match any of the vectors transmitted by the processors. In other words, the outcome of a concurrent write operation using the basic scheme is garbage.

By limiting the number of processors or the number of values that may be distinguished, however, it is possible to create a multibit CRCW memory element. For example, if we use the priority writing scheme described above, we can write any number of bits simultaneously; the price is that the number of processors can only be logarithmic in the dynamic range of the detectors. Alternatively, we can use a unary coding scheme. With this method, the number of processors can be directly proportional to the dynamic range, but the number of distinct values that can be written is only linear in the number of bits (instead of exponential). The outcome of a concurrent write operation is the median of the values being written. Note that this scheme is a generalization of the fact that the majority value is written when many processors write to a single bit.

Another approach is to perform each write operation in a number of synchronized phases that is proportional to the number of bits that are being written. All the processors that write to the same memory element simultaneously should synchronize at each phase. A simple algorithm that uses a linear number of phases and no additional hardware calls for the word to be written bit by bit. In each phase, all the participating processors write a certain bit, and then read it. Those that find a different value from the one they wrote drop out of the race, while the others continue to the next phase. The final sequence of bits is guaranteed to match the original sequence of one of the processors, thus implementing arbitrary conflict resolution.

method	*# processors*	*# values*	*# bits*	*semantics*
basic	∞	binary	1	arbitrary
basic	DR	binary	1	majority
basic	∞	2^n	n	garbage
unary	DR	$n+1$	n	median
synch phases	∞	2^n	n	arbitrary
intensities	log(DR)	2^n	n	priority

DR: dynamic range of detectors

Table 1: *Summary of the different schemes for conflict resolution.*

The various methods for conflict resolution and for writing large words are summarized in Table 1. It should be noted that more complex schemes can also be used. In fact, algorithms for the construction of multi-bit atomic registers out of single-bit ones have been a favorite topic of research for many years [26, 27, 37, 5, 11]. The above list includes only the simplest possibilities, which seem best suited for the optical implementation.

5 Conclusions

A true CRCW memory can be implemented in principle using optical technology, with small latency and no contention. However, this design is far from being practical given the current state of the art. The goal of this paper was more to expose the principles that allow concurrent access to be implemented using optics, not to suggest a specific device that should be built. The most promising characteristics of optics in this respect are:

- Optical free-space interconnections eliminate the need for shared switches that cause contention,

- Practically unlimited fan-in on the detectors allows for concurrent writes without serialization, and
- Retroreflective elements reverse the propagation of multiple independent light beams without address decoding and routing.

Much additional research must be done to deliver the promise folded in the above characteristics of optical interconnection schemes. First and foremost, there is a need to improve the characteristics of individual devices. Optical deflectors must have a short response time and a high accuracy to allow many alternative destinations to be addressed. Shutters must be fast but inexpensive. Second, a full system design is needed, keeping in mind that it must compete favorably with the capabilities of current electronic interconnection schemes. Such research will benefit mostly from a collaborative program involving both computer scientists and optical engineers and physicists.

References

[1] A. Agranat, V. Leyva, and A. Yariv, "*Voltage-controlled photorefractive effect in paraelectric* $KTa_{1-x}Nb_xO_3$*:Cu,V*". *Opt. Lett.* **14(18)**, pp. 1017–1019, Sep 1989.

[2] H. H. Barrett and S. F. Jacobs, "*Retroreflective arrays as approximate phase conjugators*". *Opt. Lett.* **4(6)**, pp. 190–192, Jun 1979.

[3] J. Beetem, M. Denneau, and D. Weigarten, "*The GF11 supercomputer*". In 12th *Ann. Intl. Symp. Computer Architecture Conf. Proc.*, pp. 108–115, Jun 1985.

[4] C. G. Bell, "*Multis: a new class of multiprocessor computers*". *Science* **228(4698)**, pp. 462–467, 26 Apr 1985.

[5] B. Bloom, "*Constructing two-writer atomic registers*". *IEEE Trans. Comput.* **37(12)**, pp. 1506–1514, Dec 1988.

[6] D. Casasent, "*Photo DKDP light valve: a review*". *Optical Engineering* **17(4)**, pp. 365–370, Jul-Aug 1978.

[7] H. J. Caulfield, "*Space-time complexity in optical computing*". In *Optical Information-Processing Systems and Architectures II*, pp. 566–572, Jul 1990. Proc. SPIE Vol. 1347.

[8] H. J. Caulfield, "*The unique advantages of optics over electronics for interconnections*". In *Advances in Interconnection and Packaging. Book Two: Microelectronic Interconnects and Packages: System and Processor Integration*, pp. 399–402, Nov 1990. Proc. SPIE Vol. 1390.

[9] R. Cole, "*Parallel merge sort*". *SIAM J. Comput.* **17(4)**, pp. 770–785, Aug 1988.

[10] M. Cronin-Golomb, B. Fischer, J. O. White, and A. Yariv, "*Theory and applications of four wave mixing in photorefractive media*". *IEEE J. Quantum Elec.* **QE-20(1)**, pp. 12–30, Jan 1984.

[11] D. Dolev and N. Shavit, "*Bounded concurrent time-stamp systems are constructible*". In 21st *Ann. Symp. Theory of Computing*, pp. 454–466, May 1989.

[12] M. M. Eshaghian, "*Parallel algorithms for image processing on OMC*". *IEEE Trans. Comput.* **40(7)**, pp. 827–833, Jul 1991.

[13] J. E. Faller and E. J. Wampler, "*The lunar laser reflector*". *Scientific American* **222(3)**, pp. 38–49, Mar 1970.

[14] J. Feinberg, "*Photorefractive nonlinear optics*". *Physics Today* **41(10)**, pp. 46–52, Oct 1988.

[15] D. G. Feitelson, *Optical Computing: A Survey for Computer Scientists.* MIT Press, 1988.

[16] D. G. Feitelson, L. Rudolph, and E. Schenfeld, "*Limitations on optical free-space crossbar-like interconnection networks*". In *Optical Interconnections and Networks*, pp. 41–52, Mar 1990. Proc. SPIE vol. 1281.

[17] S. Fortune and J. Wyllie, "*Parallelism in random access machines*". In 10th *Ann. Symp. Theory of Computing*, pp. 114–118, May 1978.

[18] A. Gottlieb, R. Grishman, C. P. Kruskal, K. P. McAuliffe, L. Rudolph, and M. Snir, "*The NYU Ultracomputer — designing an MIMD shared memory parallel computer*". *IEEE Trans. Comput.* **C-32(2)**, pp. 175–189, Feb 1983.

[19] A. Hartmann and S. Redfield, "*Design sketches for optical crossbar switches intended for large-scale parallel processing applications*". *Opt. Eng.* **28(4)**, pp. 315–327, Apr 1989.

[20] E. Hecht, *Optics*. Addison-Wesley, 2nd ed., 1987.

[21] B. A. Horwitz and F. J. Corbett, "*The PROM — theory and applications for the Pockels readout optical modulator*". *Optical Engineering* **17(4)**, pp. 353–364, Jul-Aug 1978.

[22] K. Iizuka, *Engineering Optics*. Springer-Verlag, 2nd ed., 1987.

[23] M. B. Klein and R. H. Sipman, "*Large aperture Stark modulated retroreflector at 10.8* μm". *J. Appl. Phys.* **51(12)**, pp. 6101–6104, Dec 1980.

[24] C. P. Kruskal, L. Rudolph, and M. Snir, "*Efficient synchronization on multiprocessors with shared memory*". *ACM Trans. Prog. Lang. & Syst.* **10(4)**, pp. 579–601, Oct 1988.

[25] C. P. Kruskal, M. Snir, and A. Weiss, "*The distribution of waiting times in clocked multistage interconnection networks*". *IEEE Trans. Comput.* **37(11)**, pp. 1337–1352, Nov 1988.

[26] L. Lamport, "*Concurrent reading while writing*". *Comm. ACM* **20(11)**, pp. 806–811, Nov 1977.

[27] L. Lamport, "*On interprocess communication. Part II: algorithms*". *Distributed Comput.* **1(2)**, pp. 86–101, Apr 1986.

[28] A. L. Lentine, H. S. Hinton, D. A. B. Miller, J. E. Henry, J. E. Cunningham, and L. M. F. Chirovsky, "*Symmetric self-electro-optic effect device: optical set-reset latch*". *Appl. Phys. Lett.* **52(17)**, pp. 1419–1421, Apr 1988.

[29] H. I. Mandelberg, "*Phase-modulated conjugate-wave generation in ruby*". *Opt. Lett.* **5(6)**, pp. 258–260, Jun 1980.

[30] D. M. Pepper, "*Hybrid phase conjugator/modulators using self-pumped 0°-cut and 45°-cut $BaTiO_3$ crystals*". *Appl. Phys. Lett.* **49(16)**, pp. 1001–1003, Oct 1986.

[31] D. M. Pepper, J. Feinberg, and N. V. Kukhtarev, "*The photorefractive effect*". *Scientific American* **263(4)**, pp. 34–40, Oct 1990.

[32] G. F. Pfister, W. C. Brantley, D. A. George, S. L. Harvey, W. J. Kleinfelder, K. P. McAuliffe, E. A. Melton, V. A. Norton, and J. Weiss, "*The IBM research parallel processor prototype (RP3): introduction and architecture*". In *Intl. Conf. Parallel Processing*, pp. 764–771, 1985.

[33] R. D. Rettberg, W. R. Crowther, P. P. Carvey, and R. S. Tomlinson, "*The Monarch parallel processor hardware design*". *Computer* **23(4)**, pp. 18–30, Apr 1990.

[34] A. A. Sawchuk, B. J. Jenkins, C. S. Raghavendra, and A. Varma, "*Optical crossbar networks*". *Computer* **20(6)**, pp. 50–60, Jun 1987.

[35] Y. Shiloach and U. Vishkin, "*Finding the maximum, merging and sorting in a parallel computational model*". *J. Algorit.* **2(1)**, pp. 88–102, Mar 1981.

[36] V. V. Shkunov and B. Y. Zel'dovich, "*Optical phase conjugation*". *Scientific American* **253(6)**, pp. 40–45, Dec 1985.

[37] A. K. Singh, J. H. Anderson, and M. G. Gouda, "*The elusive atomic register revisited*". In 6th *Ann. Symp. Principles of Distributed Computing*, pp. 206–221, Aug 1987.

[38] M. Snir, "*On parallel searching*". *SIAM J. Comput.* **14(3)**, pp. 688–708, Aug 1985.

[39] N. Uchida and N. Niizeki, "*Acoustooptical deflection materials and techniques*". *Proc. IEEE* **61(8)**, pp. 1073–1092, Aug 1973.

[40] U. Vishkin, "*Implementation of simultaneous memory address access in models that forbid it*". *J. Algorit.* **4(1)**, pp. 45–50, Mar 1983.

[41] T. H. Wood, C. A. Burrus, D. A. B. Miller, D. S. Chemla, T. C. Damen, A. C. Gossard, and W. Wiegmann, "*131 ps optical modulation in semiconductor multiple quantum wells (MQW's)*". *IEEE J. Quantum Elec.* **QE-21(2)**, pp. 117–118, Feb 1985.

[42] W. A. Wulf, R. Levin, and S. P. Harbison, *HYDRA/C.mmp: An Experimental Computer System.* McGraw-Hill, 1981.

[43] A. Yariv, "*Four wave nonlinear optical mixing as real time holography*". *Opt. Commun.* **25(1)**, pp. 23–25, Apr 1978.