INTERNATIONAL COMPUTER SCIENCE INSTITUTE

1947 Center Street • Suite 600 • Berkeley, California 94704 • 1-510-642-4274 • FAX 1-510-643-7684

Communicating with Low-Diffraction Lasers and Mirrors

Richard Beigel*

Yale University

TR-92-024

April 7, 1992

Abstract

Optical interconnection networks, in which each processor contains a set of lasers for communication with other processors, have long been studied. In the "regular optics" model of Murdocca [5] a bounded number of planar mirrors are used to redirect light beams, and each processor has a bounded number of lasers directed at a fixed set of angles, independent of the processor.

It is theoretically interesting to ignore diffraction, and assume that lasers beams travel in a straight line. In the regular optical model, we present elegant layouts for processor networks including the shuffle, grids, and Margulis' expander graph. We also disprove the existence of a certain kind of 3-dimensional layout for shuffles.

Using slightly more complicated optical devices, such as beam splitters, we design a "light guide," which allows simultaneous broadcasts, subject only to the limitations of light sensors. In particular, the light guide can perform single broadcasts. Given accurate enough clocks, it can perform arbitrary permutations.

*Dept. of Computer Science, 51 Prospect Street, P.O. Box 2158, Yale Station, New Haven, CT 06520-2158. Email: beigel-richard@cs.yale.edu. Supported in part by NSF grant CCR-8958528 and an NSF graduate fellowship.

1. Introduction

Good introductions to optical computing can be found in [4, 5]. Lasers can be made very small and very fast, and experimental technology is improving rapidly. At least in theory, it now appears that optical computing may scale down better than electrical computing for very large-scale integration.[1]

Current gallium-arsenide technology yields lasers several microns in diameter with switching times of several picoseconds. Unfortunately, the light from such lasers diffracts at a 30-degree angle. Gas lasers produce light beams with virtually no diffraction; however, they are too large to be useful in the production of large-scale circuits.

While current technology does not support a model of large-scale integrated optical computing in which laser beams travel in a straight line, there is already pressure to produce tiny lasers with smaller diffraction angles than are currently produced. Therefore we think it is theoretically interesting to consider optics without diffraction. After all, it is not unusual for theory to precede technology.

Previous papers have shown how to lay out processor-memory interconnects optically using lasers, light sensors, lenses, prisms, holograms, and mirrors. In this paper we will be concerned with laying out networks of processor-to-processor connections. We will lay out several processor networks using only lasers, light sensors, and mirrors. Our layouts are simpler than those for processor-memory interconnects. The mirrors redirect light beams according to the usual law of reflection: the angle of reflection equals the angle of incidence. Most of the layouts involve a small constant number of mirrors and a small constant number of lasers per processor. All processors have their lasers pointing at the same set of angles, which simplifies manufacturing. In many cases, the layouts have all their hardware on the boundary (the interior is occupied only by light beams), which may facilitate cooling.

2. Shuffles

The shuffle exchange graph[2] consists of processors numbered $0, \ldots, n$ for some odd n and two kinds of connections, called "shuffle" and "exchange." The shuffle connections are $i \rightarrow 2i \bmod n$ for $i = 1, \ldots, n-1$. The exchange connections are $i \rightarrow i + 1 \bmod n$ for $i = 0, \ldots, n$. Sometimes the graph is considered to be undirected, that is, containing the inverse of each of those connections as well.

The shuffle exchange is useful for communication, sorting, and permutations. However its layout with wires is considered impractical.

[1] It has been noted that the RC constant, which is proportional to time in the VLSI model, does not continue to decrease as wire length and width become very small. It may even increase.

[2] Some researchers prefer to number the processors from 1 to a power of 2. We find that the formulas are simpler mathematically if we number the processors starting from 0. The number of processors can be any even number.

As an interconnection network, the shuffle exchange has been implemented optically [5], though the implementation involves a beam splitter and 50% loss of energy. The shuffle exchange processor network has also been implemented with holograms [2]. We will lay out the shuffle exchange processor network elegantly, without energy loss, using mirrors rather than more complicated optics.

We will place the processors in a line. The exchange connections of the form $i \rightarrow i+1$ for $i < n$ are then trivial, though the wraparound connection $n \rightarrow 0$ takes special consideration. For the moment, we will consider only the shuffle connections.

The connections $i \rightarrow 2i$ for $0 < i < n/2$ can be performed by transmitting light perpendicular to the line of processors, and bouncing it off of a single mirror set at a 30-degree angle. The connections $i \rightarrow 2i - n$ for $n/2 < i < n$ are equivalent to $n - j \rightarrow n - 2j$ for $0 < j < n/2$, so they can be handled be a second mirror at a 30-degree angle. The layout for $n = 9$ is shown in Figure 1. Because there are no electrical components in the layout interior, this layout is easy to cool.

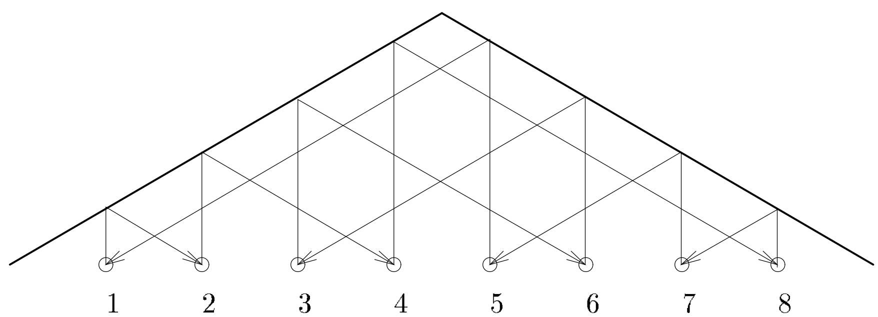

Figure 1: The shuffle connection $i \rightarrow 2i \bmod 9$

Note that each processor uses just one laser for the shuffle connection, and this laser is perpendicular to the line of processors, i.e., 90 degrees. The exchange connections, except for the wraparound, can be performed by a laser pointing directly from i to $i+1$. The inverses of the connections can be performed by pointing lasers in the direction of the incoming signals, i.e., ± 30 degrees or 0 degrees.

From a hardware designer's point of view we would prefer to dispense with processors 0 and n entirely, and do without the wraparound connection. However, we can achieve the connection by including an additional mirror and slightly altering the placement of processors 0 and n and having processor n transmit at at 150 degree angle. This is shown in Figure 2. An alternative to the third mirror would be to bounce a beam off of one of the first two mirrors at an ad hoc angle.

Another alternative is to modify the standard folding trick of VLSI design. We identify processor i with processor $n - i$ for $i = 0, \ldots, n$ and lay out the processor network as shown in Figure 3. A single processor may simulate two virtual processors by slicing up time or by including message headers to indicate which virtual processor is to receive the message.

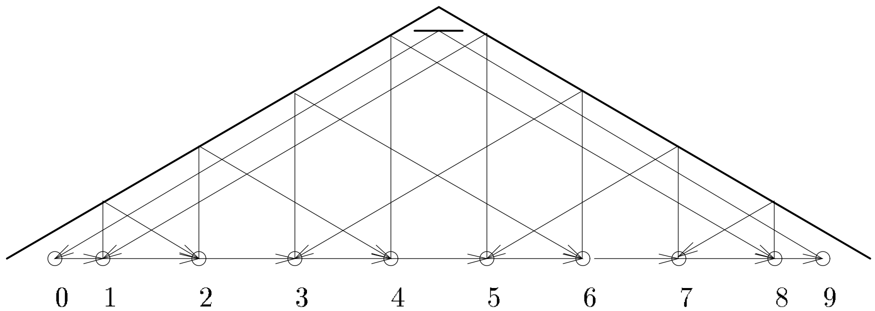

Figure 2: Including the exchange connections

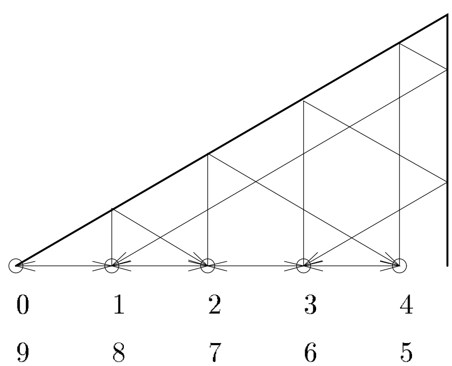

Figure 3: A folded-over shuffle exchange network

Having said more than we care to about the wraparound connections, let us reconsider the shuffle connections. The only special thing about placing mirrors at 30-degree angles is that it allows the lasers to be set at 90 degrees. Any mirror angle between 30 and 90 degrees works as well. See Figure 4. Furthermore, the layout area approaches linear as the mirror angle tends to 90 degrees. The tradeoff is that the laser beams travel closer and closer together.

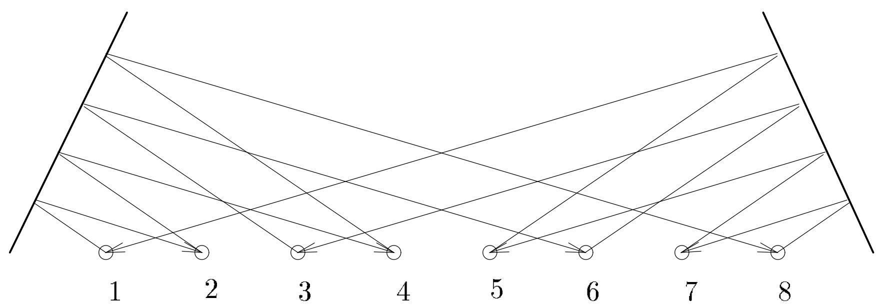

Figure 4: Larger mirror angles can reduce the layout area

3. Margulis' Expander

Expander graphs have good communication properties and are also suited to certain algorithms. Paturi et al [6] describe several applications of expanders and implement a layout in their optoelectronic model, which uses holograms. In this section we describe a regular layout using mirrors.

Margulis' expander graph contains processors numbered (x, y) for $0 \leq x, y < n$. There are uninteresting connections $(x, y) \rightarrow (x \pm 1 \bmod n, y \pm 1 \bmod n)$. The interesting connections are $(x, y) \rightarrow (x, x + y \bmod n)$ and $(x, y) \rightarrow (x + y \bmod n, y)$. We show how to lay out the $(x, y) \rightarrow (x, x + y)$ connections in 3 dimensions in Figure 5. The $(x, y) \rightarrow (x, x + y - n)$ connections can be obtained by rotating this construction 180 degrees, using a second mirror as in the shuffle example.

We can lay out the $(x, y) \rightarrow (x + y \bmod n, y)$ connections by rotating the construction above by 90 degrees. The y-direction connections can be laid out above the plane of processors, and the x-direction below. One drawback is that the processors are in the network's interior, which is not as easy to cool as the exterior.

4. Are there 3-dimensional shuffle layouts?

It would be nice to exploit three dimensions in laying out the shuffle network by placing the processors in a square grid. If the processors are numbered in row major order, then the shuffle connection is a dilation of each quadrant of the square. A

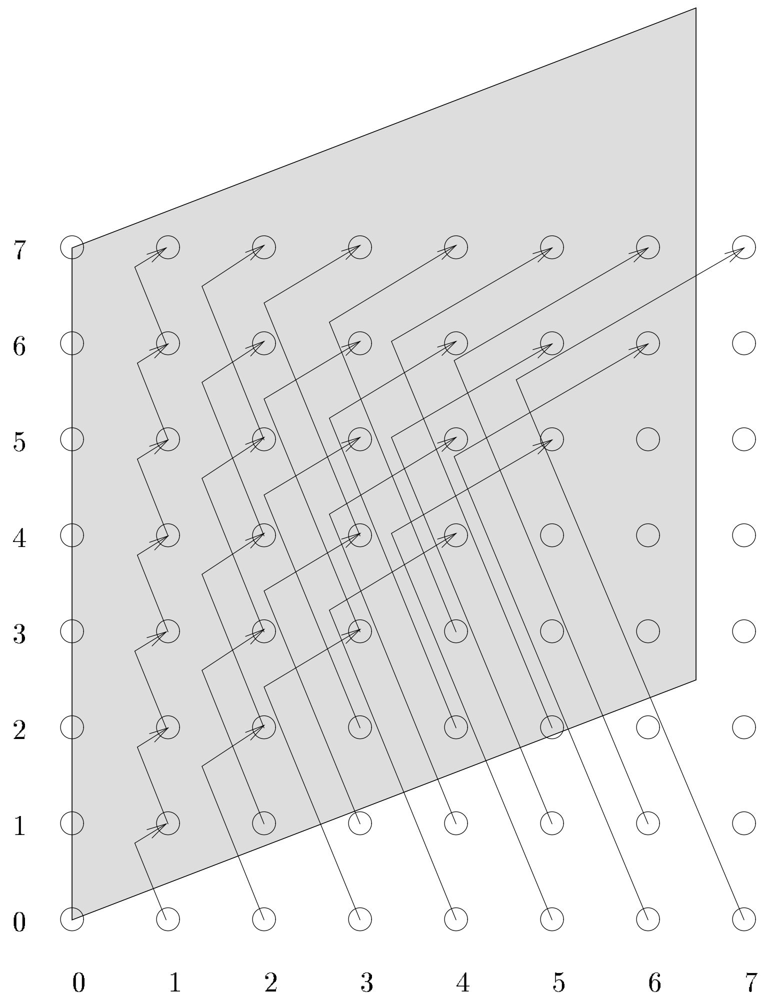

Figure 5: The $(x, y) \to (x, x + y)$ connections in Margulis' expander.

dilation can be performed using lenses or a central projection. For example, Stirk et al [8] and Lohman [3] use lenses to implement the shuffle as a processor-memory interconnect. In this section we consider whether such a layout is possible using only lasers and mirrors.

The natural 3-dimensional generalization of our 2-dimensional layout permits us to shuffle along either dimension. Thus the shuffle can be performed in two steps. We show that a single step does not suffice under reasonable restrictions. To be specific, assume the processors are at positions (x, y) for $0 < x, y < n$, and number the processors $nx + y$. We call this the *standard processor placement.* For $x, y < n/2$, the shuffle connection is $(x, y) \rightarrow (2x, 2y)$. The other 4 quadrants may be handled similarly.

If we allow light beams to pass through processors, then it is possible to lay out the connections $(x, y) \rightarrow (2x \bmod n, y/2 \bmod n)$ by shuffling in the x-direction above the plane of processors and unshuffling in the y-direction below the plane. We call this the shuffle-unshuffle network.

Using four conical mirrors, Cohen and Rajagopalan [1] have shown how to lay out $(x, y) \rightarrow (2x \bmod n, 2y \bmod n)$ with lasers pointing in the z direction. Although conical mirrors are a very reasonable extension of the model, their layout does not include the inverse connections, which would require lasers pointing in an unbounded number of directions.

Under a reasonable restriction, we show that the standard processor placement does not permit a layout of the shuffle connections. Recall that Reif [7] has shown that if light may reflect an unbounded number of times then the destination may be undecidable. Furthermore it is known that a light beam projected at a random angle in a square traces out a dense set. Thus it is reasonable for us to assume that light beams reflect a bounded ($O(1)$) number of times.

Let us classify light beams according to the initial direction and the sequence of mirrors reached. Since there are a bounded number of mirrors and a bounded number of reflections per beam, there are a bounded number of classes of beams. Consider a single class. If we reflect space though the mirrors (as it appears when we look into one) and let light beams travel straight through, then we find that every class of beams performs a parallel projection. This observation alone rules out any transformation that is not affine, i.e., that does not map lines to lines. However, it takes a bit of work to rule out nontrivial dilations.

Lemma 1. *There exists a parallel projection that maps a unit square to an $a \times b$ rectangle if and only if*

- $a \leq 1$ *or* $b \leq 1$, *and*
- $a \geq 1$ *or* $b \geq 1$.

Proof: To obtain the "if" direction, start with two planes identically oriented and perpendicular to the projection. Tilt the source place about the y axis so that projection decreases x-directional lengths by any desired factor. Tilt the destination plane about the x axis so that y-directional lengths are increased by any desired factor.

The "only if" direction can be obtained by straightforward, but tedious analytic geometry. □

Lemma 2. *Let A, B, and C be three non-collinear points. There is no parallel projection that maps the triangle ABC to a similar, but non-congruent triangle.*

Proof: A parallel projection is determined by the images of any three non-collinear points. If the projection is a non-trivial dilation when restricted to A, B, C then it is a non-trivial dilation, contradicting Lemma 1. □

Consider any three-dimensional layout of the shuffle using the standard placement. Consider only the processors (x, y) where $x, y < n/2$ so that the connection is $(x, y) \to (2x, 2y)$, a nontrivial dilation. By Lemma 2, each class must consist entirely of collinear points, so each class contains at most $n/2$ points. Therefore there must be at least $n/2$ classes.

It is an open question to find an alternative numbering of processors placed in a grid such that the shuffle connections and the exchange connections can be performed with a bounded number of mirrors and a bounded number of reflections per beam. We conjecture that this is not possible. It seems unlikely that the information-theoretic techniques used for VLSI lower bounds will be applicable here, because the two-dimensional layout is possible and also because other high-bandwidth networks like the shuffle-unshuffle can be laid out easily with the processors in a grid.

5. Multi-dimensional Grids

The optical layout of grids is fairly obvious and undoubtedly well known. If we number processors in the natural way, then for each direction in a multi-dimensional grid there is a number i such that the edges belong to the mapping $x \to x + i$. If the processors are placed in a line, this mapping is easily performed with a single mirror parallel to that line (Figure 6). One laser is required for each dimension, but the same mirror may be reused. When laying out a hypercube for example the mappings take the form $x \to x + 2^j$. A superset of the hypercube connection is shown in Figure 7. In general, each processor has one or two lasers per dimension, depending on whether the inverse connections are desired.

Three-dimensional layouts may be obtained by treating a multidimensional grid as the cross product of two lower dimensional grids, which is quite standard. Only a single planar mirror is needed.

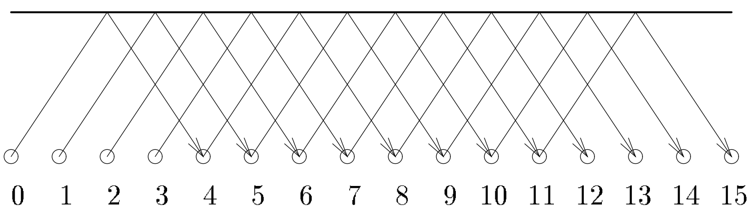

Figure 6: Laying out the connections $x \to x + 4$

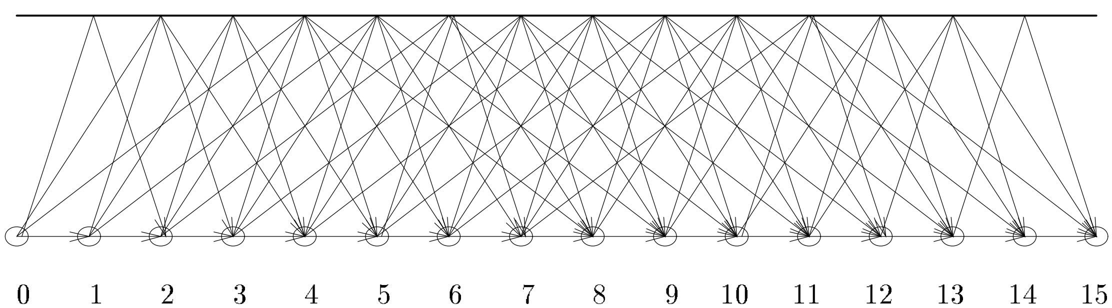

Figure 7: A superset of the 16-node hypercube.

6. The Light Guide

In this section we use slightly more interesting optical devices in order to construct much more powerful networks. A beam splitter (also called a partially silvered mirror) reflects only a fraction of the light hitting it (according to the usual rule), and it allows the remaining fraction of light to pass straight through. In Figure 8, we show a beam splitter mirror (dotted) and an ordinary mirror (solid) placed parallel to a line of processors. A single light beam from processor 1 reaches all of the processors numbered 2 through 12. If the beam splitter lets only $1/n$ of the light through then processor 2 receives almost all of the beam from processor one, and the amount reaching processors 3 through 12 is $\Theta(1/n^2)$. This disparity in signal can be alleviated by using only the odd number processors; however, $1 - 1/n$ of the energy is wasted.[3]

In similar fashion, processor i can broadcast to processors numbered $i+1, \ldots, 12$. By using the inverse connections, processor i can broadcast to processors $1, \ldots, i-1$. Thus any single broadcast is possible.

The time for a beam to travel from processor i to processor j is proportional to $|j-i|$. Thus, if processors broadcast simultaneously, each processor can determine the sender of each message based on the message's arrival time. If the processors have

[3]In order to reduce energy waste, it would be desirable if the beam splitter were more reflective on the side near the mirror than on the side near the processors. We do not know how to produce such a splitter.

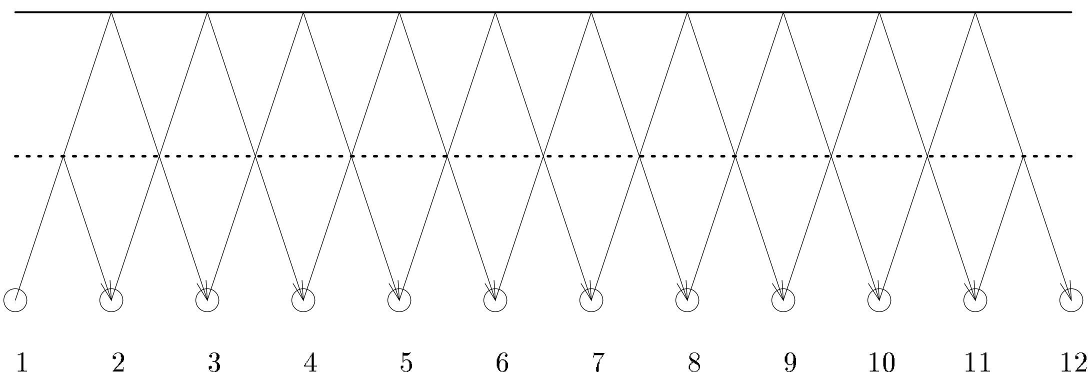

Figure 8: The Light Guide

accurate enough clocks then simultaneous broadcasts are possible. Unfortunately, light sensors may have a refractory period which prevents them from receiving closely spaced messages. It may be more reasonable to have each receiver pay attention only to messages from a single sender. Then accurate clocks permit us to perform an arbitrary permutation. The permutation can be programmed by adjusting the time at which each processor looks for a message. Hence the light guide can simulate programmable permutation networks.

7. Further Work

The networks described in this report have not yet been implemented, as far as we know. A practical obstacle to their implementation is the diffraction of light beams in free space. Because the diffraction angle is constant, beams spread out in proportion to the distance traveled. Therefore our networks do not scale up past a number of processors that is dependent on the laser technology. With current technology, this number of processors is ridiculously small. Other researchers overcome the diffraction problem by using lenses to refocus light beams. It is not clear whether that will be possible for our layouts.

Acknowledgments. This paper was born from discussions with fellow graduate student Andy Freeman at Stanford University, to whom we are greatly indebted. As usual, conversations with Manuel Blum were very helpful and encouraging. Discussions of laser beam diffraction with Charlie Stirk and Lou Guido were very helpful, as were general discussions of optical computing with Alastair McAulay, John Reif, Charlie Stirk, and Kelvin Wagner. We thank Ramamohan Paturi for suggesting that we lay out Margulis' expander graph. We are also grateful to Michael Luby, David Zuckerman, Ramamohan Paturi, Russell Impagliazzo, Steven Rudich, Stuart Russell, and Sridhar Rajagopalan for helpful discussions.

References

[1] E. Cohen and S. Rajagopalan, 1991. Personal communication.

[2] F. Kiamilev, S. C. Esener, R. Paturi, Y. Fainman, P. Mercier, C. C. Guest, and S. H. Lee. Programmable optoelectronic mutliprocessors and their comparison with symbolic substitution for digital optical computing. *Optical Engineering*, 28(4):396–409, Apr. 1989.

[3] A. W. Lohmann. What classical optics can do for the digital optical computer. *Applied Optics*, 25(10):1543–1549, May 1986.

[4] A. D. McAulay. *Optical computer architectures: the application of optical concepts to next generation computers*. Wiley, New York, 1991.

[5] M. Murdocca. *A digital design methodology for optical computing*. MIT Press, Cambridge, Mass, 1990.

[6] R. Paturi, D.-T. Lu, J. E. Ford, S. C. Esener, and S. H. Lee. Parallel algorithms based on expander graphs for optical computing. *Applied Optics*, 30(8):917–927, Mar. 1991.

[7] J. H. Reif, J. D. Tygar, and A. Yoshida. The computability and complexity of optical beam tracing. In *Proceedings of the 31st IEEE Symposium on Foundations of Computer Science*, pages 106–114. IEEE Computer Society Press, 1990.

[8] C. W. Stirk, R. A. Athale, and M. W. Haney. Folded perfect shuffle optical processor. *Applied Optics*, 27(2):202–203, Jan. 1988.