Primitive Operations for Graph-Optical Processor
(Abstract)

Shlomi Dolev* Hen Fitoussi*

April 25, 2006

Introduction. New architecture of an optical processor is suggested, an architecture that fits the special capabilities of light beams to traverse free space in parallel. Intuitively, splitting and amplifying light beams may be used to simultaneously scan many possibilities in parallel.

The location the beam reaches represents the (current) computation result. Mature technology that uses parallel light beams is already part of commercial optical communication switches. Thus, may enable us to use new approaches for coping with NP-complete problems, approach that may slightly enhance our capabilities in terms of the instances length of an NP-complete problem that we may solve in a reasonable amount of time.

Designing primitives that are designed to solve efficiently instances of TSP/Hamiltonian-Path were suggested in [1]. We extend the repertoire of NP-complete tasks solutions to include clique and independent-set.

Graph-Optical Processor Architecture. The design of this microprocessor is based on multiple, independent, identical, optical switches, which can be fully connected with each other by free space optical beams [1].

The unique feature of optical technology which we intend to explore is the ability to process many signals in parallel without wiring between the various elements. Information transfer between the elements is performed by propagating light in free space, in parallel, in a non synchronized fashion and without any cross-talk between the various signals.

Beams of lights may be initiated, split, amplified, blocked and arrive to certain locations in which threshold on the arriving number of beams (filters) may be used to identify whether the number of beams arriving to a certain location is sufficient. Detectors maybe used to detect whether a beam arrives to a certain location. In fact, filtering can be used to implement `and`/`or` logical gate in each *location*, where the detection of light following the filter indicates the output of the logical gate. In the sequel we use the term location to refer to the (3D) space coordinates and the operations (modification of the input beams to obtain the output beam that is) associated with this coordinates.

We use a semantic for locations as described next. There are two types of locations, the locations that represent the graph G and locations, that for ease of our discussion, are logically assumed to be in *columns*. Where locations in a specific *column* have the same (x, y) coordinates in space. Each column is mapped to group of states that share a specific attribute in the computation. The graph locations are mapped to each vertex $v \in V$. A programmable mask will block the light between nodes u, v such that, $(u, v) \notin E$.

Next we briefly describe the setting proposed for implementing the Hamiltonian-path, clique, and independent-set primitives.

Hamiltonian-Path. The description follows the scheme in [1]. Given a directed graph $G = (V, E)$, we would like to determine whether G contains a Hamiltonian-path. We will define a feasible path of length l as a path that consists of exactly l distinct cities.

The microprocessor is built out of $|V|$ columns, each represents a city. The i'th column contains locations which represent all feasible paths, of length j, where $1 \leq j \leq n$, that ends with city i. Light can pass between

*Department of Computer Science, Ben-Gurion University of the Negev, Beer-Sheva, 84105, Israel, {dolev,henf}@cs.bgu.ac.il. Partially supported by Intel, Ministry of Science, Deutsche Telekom, Rita Altura Trust Chair in Computer Sciences.

columns i_1 and i_2 only if $(i_1, i_2) \in E$. In-order to determine whether G contains a Hamiltonian-path we have to use detectors to check if a beam arrived to a location representing a feasible path of length $|V|$.

Clique. Given an undirected graph G, and a number k , $k \in \mathbb{N}$, we would like to determine whether there exists a set S, $S \subseteq V$, such that $|S| = k$, and S is a clique.

In the following we will describe how to determine whether a set S is a clique, using knowledge gathered on smaller size sets. Loosely speaking, given a set S, $|S| = l$, we will divide it into three roughly equal size sets, and check if each unification of every two sets is a clique of size $\mathcal{O}(\frac{2}{3}l)$. Intuitively, each possible subset of V will be represented by the (3D) space location. To each location represents a set S there are three possible incoming beams of light from three locations, which represent smaller subsets of S. S is a clique iff all three beams arrive to this location (here this location acts in a similar way to an `and` gate). In case S is not a clique, no outgoing beams will further propagate from S's location. For instance, consider the set $S = \{v_1, v_2, v_3, v_4\}$, the three possible incoming beams can arrive from the locations that represent $S_1 = \{v_1, v_2, v_3\}$, $S_2 = \{v_1, v_2, v_4\}$, and $S_3 = \{v_1, v_3, v_4\}$. In case S_1, S_2 and S_3 are not all cliques, less than three beams of light will arrive to the (3D) space location represents S, and S will not propagate light to bigger size sets.

The *fan-in* in our architecture is 3, where fan-in is the maximal number of incoming beams to a location. We prove that it is the minimal possible fan-in under an architecture that bases it's decisions on whether smaller size sets are cliques or not.

For each k, $2 \leq k \leq n$, we construct a column, of size $\binom{n}{k}$, in which each location represents a possible set S of size k. In-order to determine whether the graph contains a clique of size k, we need to check whether there exists a location, in the column suitable for k size sets, that it's fan-in equals 3. Note that lens maybe used to concentrate the outcome of the locations of a particular column to allow simple detection on the existence of such a clique.

Independent-Set. Given an undirected graph G, and a number k , $k \in \mathbb{N}$, we would like to determine whether there exists a set S, $S \subseteq V$, such that $|S| = k$, and S is an independent-set.

We can use a similar architecture to the one described for the clique problem. Given a set S we will divide it into three roughly equal size sets, and check if each unification of two sets is an independent-set of size $\mathcal{O}(\frac{2}{3}l)$. As in the previous architecture, each possible subset of V will be represented by the (3D) space location. To each location represents a set S there are three possible incoming beams of light from three locations, which represent smaller subsets of S. S is an independent-set iff all three beams did not arrive to this specific location, (this location acts as an `or` gate). Meaning, the absence of light in the location represents S will indicate that S is an independent-set. In-case S is not an independent-set, outgoing beams will further propagate from S's location and will participate in the computation of bigger size sets.

Concluding Remarks. It is known that a graph G contains an independent-set of size k iff the complementary graph $\overline{G}$ contains a clique of size k. We can use this reduction in order to use the clique architecture to solve independent-set problem, and vice versa. The Hamiltonian-path architecture has been implemented in a lab. See also [2] and [3] for recent exciting activities in the field of optical computing.

References

[1] S. Dolev, N. Yuval, "Optical implementation of bounded non-deterministic Turing machines", US Patent application #20050013531, January 2005.

[2] Lenslet LTD.

[3] H. Rong, A. Liu, R. Jones, O. Cohen, D. Hak, R. Nicolaescu, A. Fang, and M. Paniccia, "An all-sillicon Raman laser", *Nature*, 433, 292-94, (20 Jan. 2005).