Solving the Hamiltonian path problem with a light-based computer

Mihai Oltean

Department of Computer Science,
Faculty of Mathematics and Computer Science,
Babeş-Bolyai University, Kogălniceanu 1,
Cluj-Napoca, 400084, Romania.
moltean@cs.ubbcluj.ro
www.cs.ubbcluj.ro/∼moltean

Abstract. In this paper we suggest the use of light for performing useful computations. Namely, we propose a special computational device which uses light rays for solving the Hamiltonian path problem on a directed graph. The device has a graph-like representation and the light is traversing it by following the routes given by the connections between nodes. In each node the rays are uniquely marked so that they can be easily identified. At the destination node we will search only for particular rays that have passed only once through each node. We show that the proposed device can solve small and medium instances of the problem in reasonable time.

1 Introduction

Using light, instead of electric power, for performing computations is an exciting idea whose applications can be already seen on the market. This choice is motivated by the increasing number of real-world problems where the light-based devices could perform better than electric-based counterparts. Good examples in this direction can be found in the field of Optical Character Recognition [30]. Another interesting example is the n-point discrete Fourier transform which can be performed in unit time by special light-based devices [12, 24].

In this paper we suggest a new way of performing computations by using some properties of light. The idea is used within a special device for solving the Hamiltonian path problem. The device, which is very simple, has a graph-like structure. In each node there are optical cables which are used to mark (delay) the light which is passing through them. Also, there are optical cables which connect the nodes. The light is sent to the start node. In each node the ray is delayed by a certain amount of time and then is divided into several small rays which are sent to the outgoing links. At the destination node we will search for a particular ray which has passed exactly once through each node. This operation can be done easily due to the special properties of the system which delays the rays passing through a node.

The paper is organized as follows: Related work in the field of optical computing is briefly overviewed in section 2. The Hamiltonian path problem is described in section 3. The proposed device is presented in section 4. Mathematical background of the labeling system is described in sections 4.2 - 4.5. This part of the paper shows how the rays can be marked so that we can easily distinguish them. The way in which the proposed device works is given in section 4.6. A list of components required by the proposed device is given in section 4.7. Complexity is discussed in section 4.8. Suggestions for improving the device are given in sections 4.11 and 4.12. Further work directions are suggested in section 5.

2 Related work

Most of the major computational devices today are using electric power in order to perform useful computations.

Another idea is to use light instead of electrical power. It is hoped that optical computing could advance computer architecture and can improve the speed of data input and output by several orders of magnitude [9].

Many theoretical and practical light-based devices have been proposed for dealing with various problems. Optical computation has some advantages, one of them being the fact that it can perform some operations faster than conventional devices. An example is the n-point discrete Fourier transform computation which can be performed in only unit time [12, 24].

The quest for the light-based computers was started in 1929 by G. Tauschek who has obtained a patent on Optical Character Recognition (OCR) in Germany, Next step was made by Handel who obtained a patent on OCR in USA in 1933 (U.S. Patent 1,915,993). Those devices were mechanical and used templates for matching the characters. A photodetector was placed so that when the template and the character to be recognized were lined up for an exact match, and a light was directed towards it, no light would reach the photodetector. [30]. Since then, the field of OCR has grown steadily and recently has become an umbrella for multiple pattern recognition techniques (including Digital Character Recognition).

An important practical step was made by Intel researchers (see Figure 1) who have developed the first continuous wave all-silicon laser using a physical property called the Raman Effect [8, 23, 25, 26]. The device could lead to such practical applications as optical amplifiers, lasers, wavelength converters, and new kinds of lossless optical devices.

Another solution comes from Lenslet [17] which has created a very fast processor for vector-matrix multiplications (see Figure 2). This processor can perform up to 8000 Giga Multiple-Accumulate instructions per second. Lenslet technology has already been applied to data analysis using $k-$mean algorithm [19] and video compression.

A recent paper [27] introduces the idea of sorting by using some properties of light. The method called Rainbow Sort is based on the physical concepts of refraction and dispersion. It is inspired by the observation that light that

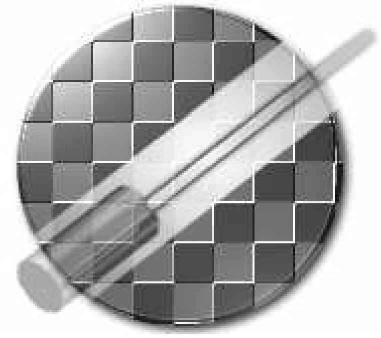

Fig. 1. Intel concept: a continuous wave laser embedded into a silicon chip.

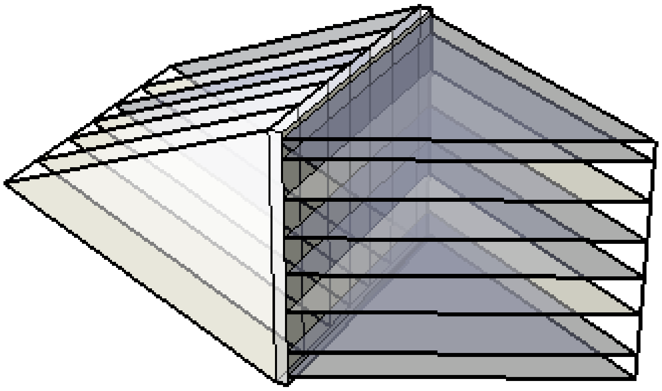

Fig. 2. A sketch of the Lenslet device used for performing vector-matrix multiplications. A vector of lasers (left part of the picture) generates the light which passes through a modulator and then is collected by an array of detectors (right part of the picture).

traverses a prism is sorted by wavelength (see Figure 3). For implementing the Rainbow Sort one need to perform the following steps:

- encode multiple wavelengths (representing the numbers to be sorted) into a light ray,
- send the ray through a prism which will split the ray into n monochromatic rays that are sorted by wavelength,
- read the output by using a special detector that receives the incoming rays.

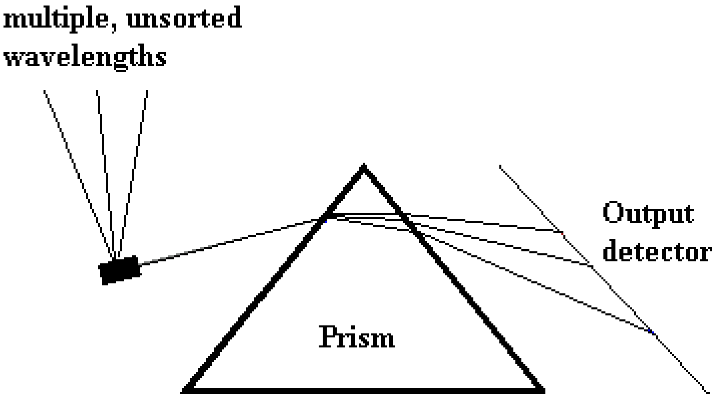

Fig. 3. Schematic view of the Rainbow Sort.

A stable version of the Rainbow Sort is proposed in [20].

Naughton (et al.) proposed and investigated [21, 28] a model called the continuous space machine which operates in discrete time-steps over a number of two-dimensional complex-valued images of constant size and arbitrary spatial resolution. The (constant time) operations on images include Fourier transformation, multiplication, addition, thresholding, copying and scaling.

There are also other devices which are taking into account the quantum properties of light. This idea has been used for solving the Traveling Salesman Problem [5, 13] using a special purpose device.

3 The Hamiltonian path problem

The description of the Hamiltonian Path problem (HPP) for a directed graph is the following:

Given a directed graph $G = (V, E)$ with $|V| = n$ nodes and a start node (v_{start}) and a stop node (v_{stop}), the problem asks to compute is there is a simple path, beginning with node v_{start} and ending with node v_{stop}, containing all nodes exactly once. The output for this decision problem is either YES or NO depending on whether the Hamiltonian path does exist or not.

The Hamiltonian path problem arises in many real-word applications [3, 7].

The problem belongs to the class of NP-complete problems [11]. No polynomial time algorithm is known for it.

A small instance of this problem was also the first problem solved using a DNA computer [1].

4 The proposed device

Our idea is based on two properties of light:

- The speed of light has a limit. The value of the limit is not very important at this stage of explanation. The speed will become important when we will compute the size of the graphs that can be solved by our device (see section 4.9). What is important now is the fact that we can delay the ray by forcing it to pass through an optical fiber cable of a certain length.
- The ray can be easily divided into multiple rays of smaller intensity/power by using some beam-splitters.

Initially a light ray is sent to the start node. Generally speaking two operations must be performed when a ray passes through a node:

- The light ray is marked uniquely so that we know that it has passed through that node.
- The ray is divided into a number of rays equal to the external degree of that node. Each obtained ray is directed toward one of the nodes connected to the current node.

At the destination node we will search only for particular rays that have passed only once through each node.

This section deeply describes the proposed system. First step is to find a way to mark the signals which passes through nodes such that the interesting signals can be easily identified at the destination node. The mathematical background required for this operation is described in sections 4.1 - 4.5 and the physical implementation of the labeling system is described in section 4.7.

4.1 Labeling system

At the destination node we will wait for a particular ray which has passed through all nodes of the graph exactly once. This is why we need to find a way to label that particular ray so that it could be easily identified.

Actually we are interested in marking all rays which pass through a particular node with a unique label, such that Hamiltonian path is uniquely identified at the destination node (v_{stop}).

In the solution proposed in this paper, the rays passing through a node are marked by delaying them with a certain amount of time. This delay can be easily obtained by forcing the rays to pass through an optical fiber of a certain length.

Roughly speaking, we will know if a certain ray has traversed a Hamiltonian path only if its delay (at the destination node) is equal to the sum of delays of all nodes in that graph. We will also know the particular moment when the expected ray (the one which has completed a Hamiltonian path) will arrive. In this case the only thing that we have to do is to "listen" if there is a fluctuation in the intensity of the signal at that particular moment. Due to the special properties of the proposed system we know that no other ray will arrive, at the destination node, at the moment when the Hamiltonian path ray has arrived.

The delays, which are introduced by each node, cannot take any values. If we would put random values for delays we might have different rays (which are not Hamiltonian paths) arriving, at the destination node, in the same time with a ray representing a Hamiltonian path.

We need only the ray, which has traversed a Hamiltonian path, to arrive in the destination node at the moment equal to the sum of delays of each node (the moment when the ray has entered in the start node is considered moment 0). Thus, the delaying system must have the following property:

Property of the delaying system

Let us denote by d_1, d_2, ..., d_n the delays introduced by each node of the graph. A correct set of values for this system must satisfy the condition:

$d_1 + d_2 + ... + d_n \neq a_1 \cdot d_1 + a_2 \cdot d_2 + ... + a_n \cdot d_n$,

where a_i $(1 \leq i \leq n)$ are natural numbers $(a_i \geq 0)$ and cannot be all 1 in the same time.

Basicaly speaking a_k tell us how many times the ray has passed through node k. Thus, if value a_k is strictly greater than 1 it means that the ray has passed at least twice through node k.

Finding the appropriate labeling system was a three-step process. First of all we have written a computer program which generates this numbers by using a backtracking procedure [6]. Then we have extracted a general formula and we have proved its correctness.

4.2 Backtracking procedure

We also wanted to generate numbers such that the highest number in a system is the smallest possible. This will ensure that the network is constructed in an efficient way. The labeling systems generated by our computer programs are given in Table 1.

4.3 Extracting the general formula

From Table 1 it can be easily seen that these numbers follow a general rule. For a graph with n nodes one of the possible labeling systems is:

$2^n - 2^{n-1}$,
$2^n - 2^{n-2}$,

Table 1. The labeling system generated by our backtracking procedure. First column contains the number of nodes of the graph. The second column represents the labels applied to nodes.

n	Labels (delays)
1	1
2	2, 3
3	4, 6, 7
4	8, 12, 14, 15
5	16, 24, 28, 30, 31
6	32, 48, 56, 60, 62, 63

$2^n - 2^{n-3}$,
... ,
$2^n - 2^0$.

Remarks

- The numbers in this set are also called Nialpdromes (sequence A023758 from The On-line Encyclopedia of Integer Numbers [29]). They are numbers whose digits in base 2 are in nonincreasing order.
- These numbers have been used in [16] for solving NP-complete problems in the context of DNA computers [1],
- The delaying system described above has the advantage of being a general one, but it also has a weakness: it is exponential.

4.4 Proving the correctness

We have to prove that the property of delaying system (see section 4.1) holds for this sequence of numbers.

Actually we have to prove that the equality:

$$\begin{aligned} 2^n - 2^{n-1} + 2^n - 2^{n-2} + 2^n - 2^{n-3} + ... + 2^n - 2^0 = \\ a_1 \cdot (2^n - 2^{n-1}) + a_2 \cdot (2^n - 2^{n-2}) + a_3 \cdot (2^n - 2^{n-3}) + ... + a_n \cdot (2^n - 2^0) \end{aligned} \quad (1)$$

(where $a_i \geq 0$) is not possible unless all a_i are equal to 1.

The left part of the equality is:

$2^n - 2^{n-1} + 2^n - 2^{n-2} + 2^n - 2^{n-3} + ... + 2^n - 2^0 =$
$n \cdot 2^n - (2^{n-1} + 2^{n-2} + 2^{n-3} + ... + 2^0) =$
$n \cdot 2^n - 2^n + 1 =$

$(n-1) \cdot 2^n + 1$.

First of all we have to see that the equality does not hold if all a_i numbers are at least 1 and at least one number is strictly greater than 1. If this happens the 2^n term will be represented at least n times. But, it must be represented only $n-1$ times (see above).

Thus, if at least one number a_i is strictly greater than 1, it means that other numbers a_j must be 0. We will prove that the equality (1) does not hold in this case too.

As discussed above, if one of the coefficients is 0, at least one of the other coefficients must be strictly greater than 1. For instance, if a_1 is set to 0, it means that a_2 must be set to 3 in order to compensate the missing 2^{n-1} term. This is a direct consequence of the fact that $2^{n-1} = 2 \cdot 2^{n-2}$. Of course, we also have to take into account that 2^{n-2} must be represented once. This is why we have to set a_2 to value 3.

If a_2 is also 0 we have to set a_3 to value 7 (we need $4 \cdot 2^{n-3}$ in order to compensate the missing term 2^{n-1}, we also need $2 \cdot 2^{n-3}$ to compensate the missing term 2^{n-2} and, of course, the term 2^{n-3} must be represented 1 time).

As a general idea: if a particular term (2^{n-j}) is missing (the corresponding coefficient a_j is set to 0), it can be compensate by setting one of the next coefficients to a value of at least 3. But, a coefficient set to 0 means that the term 2^n is missing once, and by setting another coefficient to at least 3 we will get at least 2 extra representations for 2^n. This will mean that right part of the equation (1) will be at least $(n+1) \cdot 2^n + 1$. But, the left part of the equation is only $(n-1) \cdot 2^n + 1$.

With this we have shown that all $a_i (1 \leq i \leq n)$ must be 1 in order to have equality in equation (1). For any other values of a_i the equality (1) does not hold.

4.5 Proving the minimality

An important question is whether this system is the minimal possible (the greatest number is the minimal possible). We prove this by showing a counterexample.

Let us suppose that there is another delaying system which is smaller than the one described in section 4.3. This means that the largest delay is smaller than $2^n - 2^0$. We are not interested in the other $(n-1)$ delays because the largest one imposes the largest length for the cables.

Consider a graph with 2 nodes. In this case the delaying system is $\{2, 3\}$. We will show that the largest delay cannot be smaller than 3.

If it is smaller it should be 2 or 1 (actually it cannot be 1 because, in this case, the other delay should be 0 and we do not allow this delay in our system). If the delay is 2 it means that the delay for the other node is 1. The sum of the delays is 3 which can be simply obtained by visiting 3 times the node with delay 1. This is in conflict with the property of the delaying system (see section 4.1).

With this we have proved that the largest delay cannot be smaller than $2^n - 2^0$.

4.6 How the system works

An schematic example of a graph-like system is given in Figure 4.

In the graph depicted in Figure 4 the light will enter in node 0. It will be delayed with a certain amount of time and then it will be divided into 3 rays which will be sent to nodes 1, 3 and 6. In node 1 the ray will be delayed (with the amount of time corresponding to node 1) and then it will be sent to nodes 2 and 3. One of the rays will be able to generate and Hamiltonian path (0, 1, 2, 3, 4, 5, 6).

Note that there are also several cycles in the graph: 1, 3, 4 or 1, 2 or 2, 3. The cycles will make that some particular rays to be trapped within the system. The rays which have passed once through the previously described cycle are not considered Hamiltonian paths because the moments when they arrive, at the destination node, are different from the sum of delays introduced by each node.

Because we are working with continuous signal we cannot expect to have discrete output at the destination node. This means that rays arrival is notified by fluctuations in the intensity of the light. These fluctuations will be transformed, by the photodiode, in fluctuations of the electric power which will be easily read by the oscilloscope.

4.7 Physical implementation of the labeling system

For implementing the proposed device we need the following components:

- a source of light (laser),
- Several beam-splitters for dividing light rays into multiple subrays. A standard beam-splitter is designed using a half-silvered mirror. For dividing a ray into k subrays we need $k - 1$ beam-splitters. An example on how to split a ray in more than 2 subrays is given in Figure 5,
- A high speed photodiode for converting light rays into electrical power. The photodiode is placed in the destination node,
- A tool for detecting fluctuations in the intensity of electric power generated by the photodiode (oscilloscope),
- A set of optical fiber cables having certain lengths. These cables are used for connecting nodes and for delaying the signals within nodes. The length of the cables must obey the rules described in section 4.1. A practical example on how to compute their length is given in section 4.9.

4.8 Complexity

This section answers a very important question: *Why the proposed approach is not a polynomial-time solution for the HPP?*

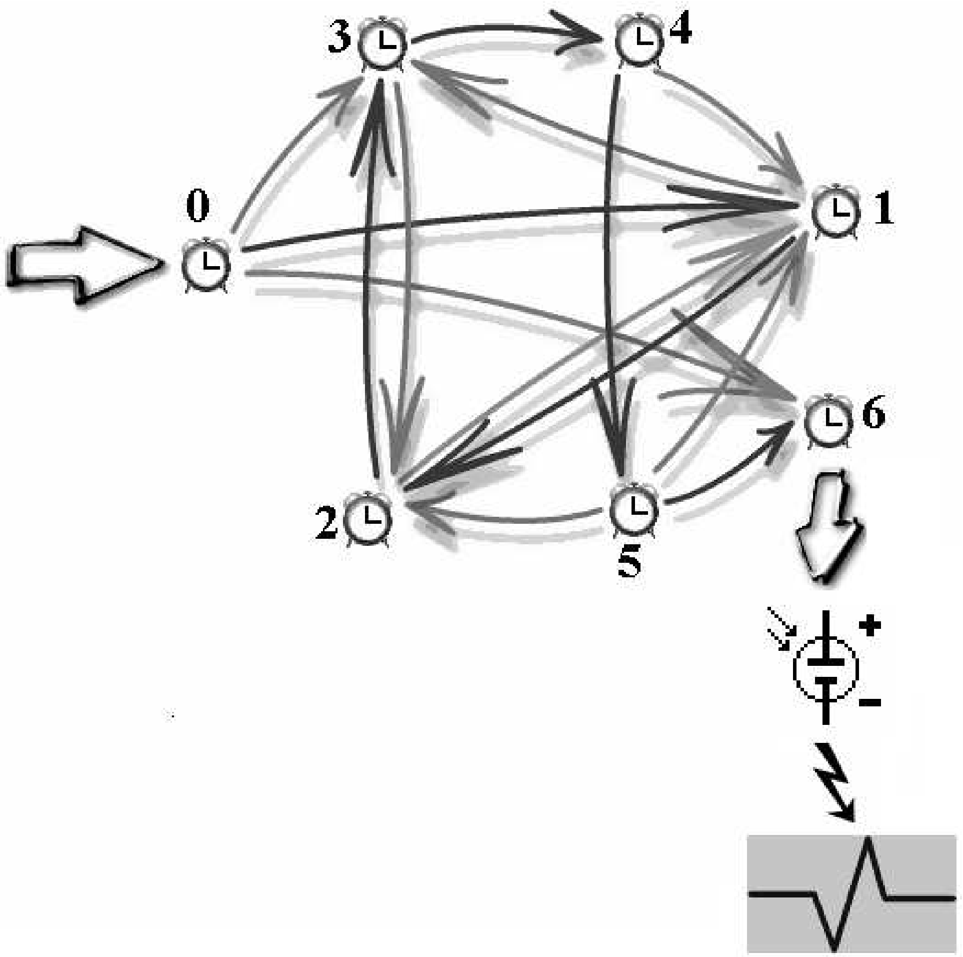

Fig. 4. A schematic representation of the proposed device. Consider a graph with 7 nodes. Start node is 0 and the destination node is 6. The list of arcs is: (0, 1), (0, 3), (0, 6), (1, 2), (1, 3), (2, 1), (2, 3), (3, 2), (3, 4), (4, 1), (4, 5), (5, 2), (5, 6). In each node the rays are delayed with a certain amount of time. At the destination node we will have a photodiode and an oscilloscope. When a ray will arrive in the destination node there will be a fluctuation, in the light intensity, which could be measured by the oscilloscope.

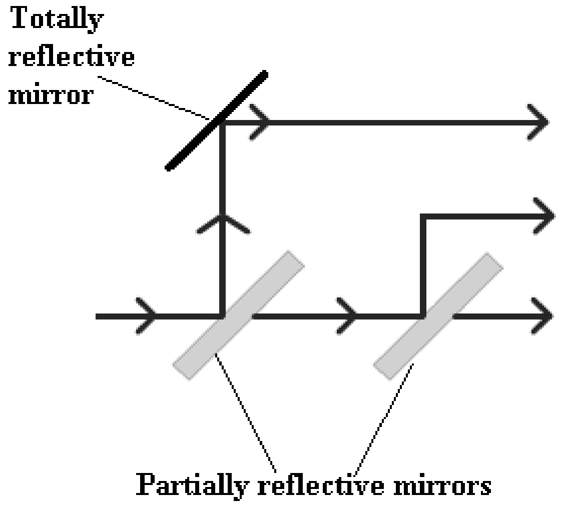

Fig. 5. The way in which a ray can be split into 3 sub-rays by using 2 beam-splitters.

At the first sight one may be tempted to say that the proposed approach provides a solution in polynomial time to any instance of the HPP. The reason behind such claim is given by the ability of the proposed device to provide output to any instance by traversing only once all nodes (O(n) complexity). This could mean that we have found a polynomial-time algorithm for the HPP. A direct consequence is obtaining solutions, in polynomial time, for all other NP-Complete problems - since there is a polynomial reduction between them [11].

However, this is not our case. As can be seen from Table 1 the delay time increases exponentially with the number of nodes. Even if the ray has to traverse only n nodes (resulting a complexity of O(n)), the total time required by the ray to reach the destination node increases exponentially with the number of nodes.

There are two direct consequences which are derived from here:

- The length of the optical fibers, used for delaying the signals, increases exponentially with the number of nodes,
- The intensity of the signal decreases exponentially with the number of nodes that are traversed.

These two issues are discussed in sections 4.9 and 4.10.

4.9 Problem size

We are interested in computing the size of the cables required to solve a certain instance of the problem in a small amount of time. This will give as a rough indication on the size of the graphs that can be solved using our system in reasonable time.

This size heavily depends on the accuracy of the measurement tools (response time of the photodiode and the rise time of the oscilloscope).

The rise-time of the best oscilloscope available on the market is in the range of picoseconds (10^{-12} seconds). This means that we should not have two signals that arrive at 2 consecutive moments at a difference smaller than 10^{-12} seconds.

Knowing that the speed of light is $3 \cdot 10^8 m/s$ we can easily compute the minimal cable length that should be traversed by the ray in order to be delayed with 10^{-12} seconds. This is obviously 0.0003 meters.

This value is the minimal delay that should be introduced by a node in order to ensure that the difference between the moments when two consecutive signals arrive at the destination node is greater or equal to the measurable unit of 10^{-12} seconds. This will also ensure that we will be able to correctly identify whether the signal has arrived in the destination node at a moment equal to the sum of delays introduced by each node. No other signals will arrive within a range of 10^{-12} seconds around that particular moment.

Once we have the length for that minimal delay is quite easy to compute the length of the other cables that are used in order to induce a certain delay.

Recall from section 4.1, Table 1 that a graph with 5 nodes has the following delaying system:

16, 24, 28, 30, 31.

From the previous reasoning line we have deduced that the smaller indivisible unit is 0.0003. So, we have to multiply these numbers by 0.0003. We obtain:

0.00048, 0.00072, 0.00084, 0.0009, 0.00093.

These numbers represent the length of the cables that must be used in graph's nodes in order to induce a certain delay.

Note that the delay introduced by the cables connecting the nodes was not taken into account in this example. This is not a limitation of our system. The cables connecting nodes can be set to have some length which must obey the property of delaying system (see section 4.1). Note that all cables must have the same length. In this case if we have a graph with 4 nodes the length of every cable connecting the nodes must be set to 16 units (the shortest possible - in order to reduce the costs). The length of cables within the nodes should be 24, 28, 30 and 31 units.

The largest length in this sequence is 0.00093 meters. This length is not very big, but for larger graphs the length of the cables within nodes can be a problem.

Once we have the length for that minimal delay is quite easy to compute the maximal number of nodes that a graph can have in order to find the Hamiltonian path in one second. We know the facts:

- the largest delay has the form $2^n - 1$ (see equation 1),

- the distance traversed by light in 1 second is $3 \cdot 10^8$ meters,
- the shortest delay possible is 0.0003 meters.

We simply have to solve the equation:

$$2^n \cdot 0.0003 = 3 \cdot 10^8 \tag{2}$$

This number is about 40 nodes. However, the length of the optic fibers used for inducing the largest delay for this graph is huge: about $8 \cdot 10^8$ meters. We cannot expect to have such long cables for our experiments.

However, shorter cables (of several hundreds of kilometers) are already available in the internet networks. They can be easily used for our purpose. Assuming that the longest cable that we have is about 300 kilometers we may solve instances with about 26 nodes. The amount of time required to obtain a solution is about 10^{-6} seconds.

Note that the maximal number of nodes can be increased when the precision of our measurement instruments (oscilloscope and photodiode) is increased.

Also note that this difficulty is not specific to our system only. Other major unconventional computation paradigms, trying to solve NP-complete problems share the same fate. For instance, a quantity of DNA equal to the mass of Earth is required to solve HPP instances of 200 cities using DNA computers [14].

4.10 Amplifying the signal

Beam splitters are used in our approach for dividing a ray in two or more subrays. Because of that, the intensity of the signal is decreasing. In the worst case we have an exponential decrease of the intensity. For instance, in a graph with n nodes, each signal is divided (within each node) into $n-1$ signals. Roughly speaking, the intensity of the signal will decrease n^n times.

This means that, at the destination node, we have to be able to detect very small fluctuations in the intensity of the signal. For this purpose we will use a photomultiplier [10] which is an extremely sensitive detector of light in the ultraviolet, visible and near infrared range. This detector multiplies the signal produced by incident light by as much as 10^8, from which even single photons can be detected.

4.11 Improving the device by reducing the speed of the signal

The speed of the light in optic fibers is an important parameter in our device. The problem is that the light is too fast for our measurement tools. We have either to increase the precision of our measurement tools or to decrease the speed of light.

It is known that the speed of light traversing a cable is significantly smaller than the speed of light in the void space. Commercially available cables have limit the speed of the ray wave up to 60% from the original speed of light. This means that we can obtain the same delay by using a shorter cable.

However, this method for reducing the speed of light is not enough for our purpose. The order of magnitude is still the same. This is why we have the search for other methods for reducing that speed. A very interesting solution was proposed in [15] which is able to reduce the speed of light by 7 orders of magnitude and even to stop it [4, 18]. In [4] they succeeded in completely halting light by directing it into a mass of hot rubidium gas, the atoms of which, behaved like tiny mirrors, due to an interference pattern in two control beams.

This could help our mechanism significantly. However, how to use this idea for our device is still an open question because of the complex equipment involved in those experiments [15, 18].

By reducing the speed of light by 7 orders of magnitude we can reduce the size of the involved cables by a similar order (assuming that the precision of the measurement tools is still the same). This will help us to solve larger instances of the problem.

4.12 Improving the performance of the device for particular graphs

The labeling system proposed in section 4.1 is a general one. It can be used for any kind of graph (with any number of nodes and arcs). We have shown that this system has a big problem: the value of the involved numbers increase exponentially with the number of nodes in the graph being solved.

But, for particular graphs we can find other labeling systems which are not exponential. For example, the linear graph (see Figure 6 can be solved by our device by using virtually no delays. In this case the moment when the signal arrives in the destination node is equal to sum of delays introduced by the cables connecting the nodes.

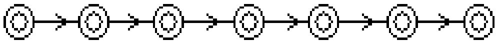

Fig. 6. A linear graph with 7 nodes. No delays are required for the nodes of this graph in order to find a Hamiltonian path

Finding the optimal labeling system for a particular graph is an interesting problem which will be investigated in the near future.

4.13 Technical challenges

There are many technical challenges that must be solved when implementing the proposed device. Some of them are:

- Cutting the optic fibers to an exact length with high precision. Failing to accomplish this task can lead to errors in detecting a ray which has passed through each node once.
- Finding a high precision oscilloscope. This is an essential step for solving larger instances of the problem (see section 4.9),
- Finding cables long enough so that larger instances of the problem could be solve. This problem might have a simple solution: the internet networks connecting the world cities. It is easy to find cables of hundreds of kilometers connecting distant cities. This will help us to solve instance of more than 10 nodes. However, this solution introduces a difficulty too: cables of certain lengths must be found or the system must be rescaled in order to fit the existing lengths.

5 Conclusions and further work

The way in which light can be used for performing useful computations has been suggested in this paper. The techniques are based on the massive parallelism of the light ray.

It has been shown the way in which a light-based device can be used for solving the Hamiltonian path problem. Using the today technology we can build a light-based device which can solve small and medium size instances in several seconds.

Further work directions will be focused on:

- Implementing the proposed device,
- Finding optimal labeling systems for particular graphs. This will reduce the length of the involved cables significantly,
- Our device cannot find the Hamiltonian path. It can only say if this path exists or not. If there are multiple paths we cannot distinguish them. However, the HPP YES/NO decision problem is still a NP-complete problem [11]. We are currently investigating a way to store the order of nodes so that we can easily reconstruct the path,
- Finding other non-trivial problems which can be solved by using the proposed device,
- Finding other ways to introduce delays in the system. The current solution requires cables that are too long and too expensive,
- Using other type of signals instead of light. Possible candidates are electric power and sound,
- Finding other ways to implement the system of marking the signals which pass through a particular node. This will be very useful because the currently suggested system, based on delays, is too time consuming. The other properties of light can be used for this purpose: amplitude, wavelength, etc.

Acknowledgments

The author likes to thanks to the anonymous reviewers and to the participants to Unconventional Computing 2006 conference (specially to Mikhail Prokopenko, Damien Woods, Jerzy Górecki, Alexis De Vos, Russ Abbott, William Langdon, Pierluigi Frisco) for providing useful suggestions on an earlier version of this paper [22].

References

1. Adleman L (1994) Molecular computation of solutions to combinatorial problems. Science 266:1021-1024
2. Agrawal GP (2002) Fiber-optic communication systems. Wiley-Interscience; 3^{rd} edition
3. Ascheuer N (1995) Hamiltonian path problems in the on-line optimization of flexible manufacturing systems. PhD thesis, TU Berlin
4. Bajcsy M, Zibrov AS, Lukin MD (2003) Stationary pulses of light in an atomic medium. Nature 426:638-641.
5. Černý V (1993) Quantum computers and intractable (NP-Complete) computing problems. Phys. Rev. A, 48:116-119
6. Cormen TH, Leiserson CE, Rivest RR (1990) Introduction to algorithms. MIT Press
7. Doniach S, Garel H, Orland H, (1996) Phase diagram of a semiflexible polymer chain in a θ solvent: Application to protein folding. Journal Of Chemical Physics, 105:1601-1608
8. Faist J (2005) Optoelectronics: silicon shines on. Nature, 433:691-692
9. Feitelson DG (1988) Optical computing: A survey for computer scientists, MIT Press
10. Flyckt SO, Marmonier C (2002) Photomultiplier tubes: Principles and applications. Photonis, Brive, France
11. Garey MR, Johnson DS (1979) Computers and intractability: A guide to NP-Completeness. Freeman & Co, San Francisco, CA
12. Goodman JW (1982) Architectural development of optical data processing systems. Aust. J. Electr. Electron. Eng. 2:139149
13. Greenwood GW (2001) Finding solutions to NP problems: Philosophical differences between quantum and evolutionary search algorithms. in Proceedings CEC'2001, IEEE Press, 815-822
14. Hartmanis J (1995) On the weight of computations. Bulletin of the EATCS 55:136-138
15. Hau LV, Harris SE, Dutton Z, Behroozi CH, (1999) Light speed reduction to 17 meters per second in an ultracold atomic gas. Nature 397:594-598
16. Henkel C, Frisco P, Tengely Sz (2005) An algorithm for SAT without an extraction phase. DNA Computing, Eleventh International Meeting on DNA Based Computers, LNCS 3892, in press.
17. Lenslet website, www.lenslet.com
18. Liu C, Dutton Z, Behroozi CH, Hau LV (2001) Observation of coherent optical information storage in an atomic medium using halted light pulses. Nature 409:490-493
19. MacQueen J (1967) Some methods for classification and analysis of multivariate observations. In LeCam LM, Neyman J, (eds.) Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability. University of California press, Berkeley, 281-297

20. Murphy N, Naughton TJ, Woods D, Henley B, McDermott K, Duffy E, van der Burgt PJM, Woods N (2006) Implementations of a model of physical sorting. From Utopian to Genuine Unconventional Computers workshop, Adamatzky A, Teuscher C, (editors), Luniver Press 79-100
21. Naughton TJ (2000) A model of computation for Fourier optical processors. In Lessard RA, Galstian T (editors), Optics in Computing, Proc. SPIE 4089:24-34
22. Oltean M (2006) A light-based device for solving the Hamiltonian path problem. Unconventional Computing, Calude C. (et al.) (Eds), LNCS 4135, Springer-Verlag, 217-227
23. Paniccia M, Koehl S (2005) The silicon solution. IEEE Spectrum, IEEE Press, October
24. Reif JH, Tyagi A (1997) Efficient parallel algorithms for optical computing with the discrete Fourier transform primitive. Applied optics 36(29):7327-7340
25. Rong H, Jones R, Liu A, Cohen O, Hak D, Fang A, Paniccia M (2005) A continuous-wave Raman silicon laser. Nature 433:725-728
26. Rong H, Liu A, Jones R, Cohen O, Hak D, Nicolaescu R, Fang A, Paniccia M (2005) An all-silicon Raman laser. Nature 433:292-294
27. Schultes D (2005) Rainbow Sort: Sorting at the speed of light. Natural Computing. Springer-Verlag, 5(1):67-82
28. Woods D, Naughton TJ (2005) An optical model of computation. Theoretical Computer Science, 334 (1-3):227-258
29. Sloane N, The on-line encyclopedia of integer sequences. $http://www.research.att.com/\sim njas/sequences/A$023758
30. Optical Character Recognition @ Wikipedia, $http://en.wikipedia.org/wiki/Optical\_character\_recognition$