Solving the subset-sum problem with a light-based device

Mihai Oltean, Oana Muntean

Department of Computer Science,
Faculty of Mathematics and Computer Science,
Babeş-Bolyai University, Kogălniceanu 1,
Cluj-Napoca, 400084, Romania.
mihai.oltean@gmail.com
oana_muntean85@yahoo.com
`https://mihaioltean.github.io/optical`

Abstract. We propose a special computational device which uses light rays for solving the subset-sum problem. The device has a graph-like representation and the light is traversing it by following the routes given by the connections between nodes. The nodes are connected by arcs in a special way which lets us to generate all possible subsets of the given set. To each arc we assign either a number from the given set or a predefined constant. When the light is passing through an arc it is delayed by the amount of time indicated by the number placed in that arc. At the destination node we will check if there is a ray whose total delay is equal to the target value of the subset sum problem (plus some constants).

1 Introduction

Unconventional computing means computing by using new or unusual methods. Special properties of matter are usually exploited in an unconventional manner. Several unconventional techniques and approaches for attacking difficult problems have been investigated so far: DNA computing [2], Quantum computing [28], Soap bubbles [1, 6], Gear-based computers [29], Adiabatic algorithms [16] etc.

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 [31]. Another interesting example is the n-point discrete Fourier transform which can be performed in unit time by special light-based devices [13, 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 subset-sum problem. The problem asks to find if there is a subset of given set A whose sum is B.

The device, which is very simple, has a graph-like structure. The nodes are connected by arcs in such way all possible subsets of A are generated. To each arc we assign either a number from the given set A, or a predefined constant. The length of an arc is directly proportional to the number assigned to it.

Initially a light ray is sent to the start node. In each node the light is divided into 2 subrays. Each arc delays the ray by an amount of time equal to the number assigned to it. At the destination node we will check if there is a ray arriving in the destination node at the moment equal to the target value of B (plus some constants introduced by the system).

The paper is organized as follows: Related work in the field of optical computing is briefly overviewed in section 2. The subset-sum problem is described in section 3. The proposed device is presented in section 4. The way in which the proposed device works is given in section 4.4. A list of components required by the proposed device is given in section 5. Complexity is computed in section 6. Weaknesses of our device are discussed in section 7. Section 7.2 gives a rough approximation for the size of the instances that can be solved by our device. Suggestions for improving the device are given in section 7.5. Further work directions are suggested in section 8.

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, optically, in only unit time [13, 24]. Based on that, a solution to the subset sum problem can be obtained by discrete convolution. The idea is that the convolution of 2 functions is the same as the product of their frequencies representation [32].

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. 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 [31]. 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 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 1 (a)). 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 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 traverses a prism is sorted by wavelength (see Figure 1 (b)). 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.

A stable version of the Rainbow Sort is proposed in [19].

Naughton (et al.) proposed and investigated [20, 30] 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.

A system which solves the Hamiltonian path problem (HPP) [11] by using light and its properties has been proposed in [21, 22]. The device has the same structure as the graph where the solution is to be found. The light is delayed within nodes, whereas the delays introduced by arcs are constants. Because the problem asks that each node has to be visited exactly once, a special delaying system was designed. At the destination node we will search for a ray which has visited each node exactly once. This is very easy due to the special properties of the delaying system.

3   The subset-sum problem

The description of the subset-sum problem [7, 11] is the following:

Given a set of positive numbers $A = \{a_1, a_2, ..., a_n\}$ and another positive number B. Is there a subset of A whose sum equals B?

We focus our attention on the YES / NO decision problem. We are not interested in finding the subset generating the solution. Actually we are interested to find only if such subset does exist.

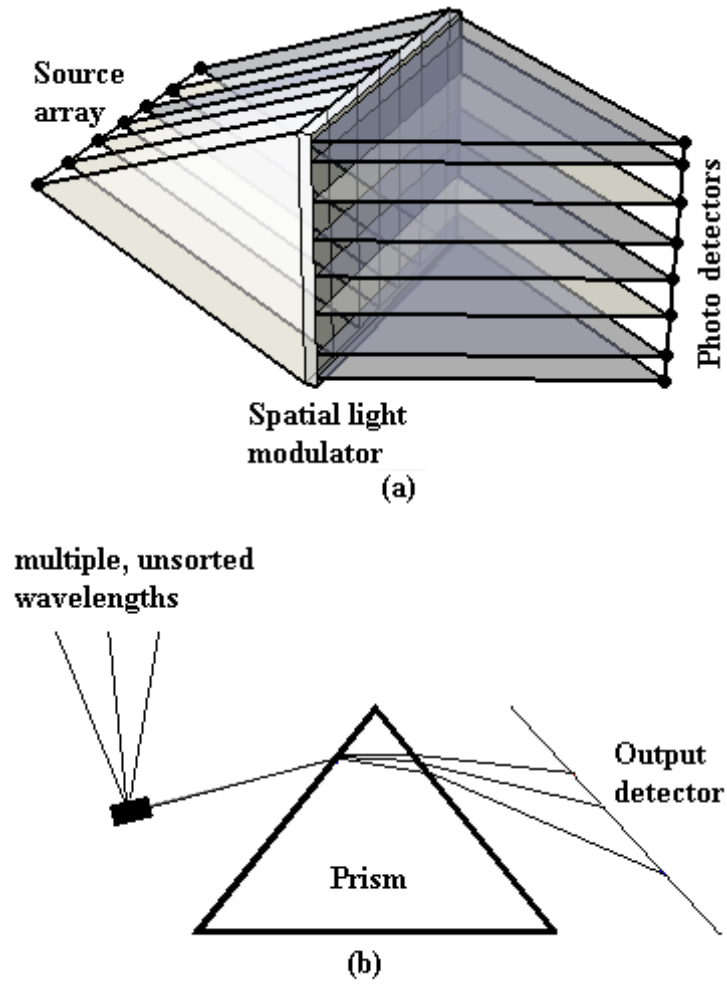The subset-sum problem arises in many real-word applications [12].

Fig. 1. (a) A sketch of the Lenslet device used for performing vector-matrix multiplications. (b) Schematic view of the Rainbow Sort.

The problem belongs to the class of NP-complete problems [11]. No polynomial time algorithm is known for it. However, a pseudo-polynomial time algorithm does exist for this problem [11]. The complexity of this algorithm is bounded by both n and B. The algorithm requires $O(n * B)$ storage space.

4 The proposed device

This section deeply describes the proposed system. Section 4.1 describes the properties of light which are useful for our device. Section 4.2 introduces the operations performed by the components of our device. Basic ideas behind our concept are given in section 4.3. Some examples on how the system works are given in section 4.4.

4.1 Useful properties of light

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 try to measure the moment when rays arrive at the destination node (see section 7.1). 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. Beam-splitters are used for this operation [3, 9].

4.2 Operations performed within our device

The proposed device has a graph like structure. Generally speaking one operation is performed when a ray passes through a node and one operation is performed when a ray passes through an edge.

 – When passing through an arc the light ray is delayed by the amount of time assigned to that arc.
 – When the ray is passing through a node it 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.

4.3 The device

The first idea for our device was that numbers from the given set A represent the delays induced to the signals (light) that passes through our device. For instance, if numbers a_1, a_3 and a_7 generate the expected subset, then the total delay of the signal should be $a_1 + a_3 + a_7$. If using light we can easily induce some delays by forcing the ray to pass through an optical cable of given length.

This is why we have designed our device as a directed graph. Arcs, which are implemented by using optical cables, are labeled with numbers from the given

set A. Each number is assigned to exactly one arc and there are no two arcs having assigned the same number. There are $n+1$ nodes connected by n arcs. At this moment of explanation we have a linear graph as the one shown in Figure 2.
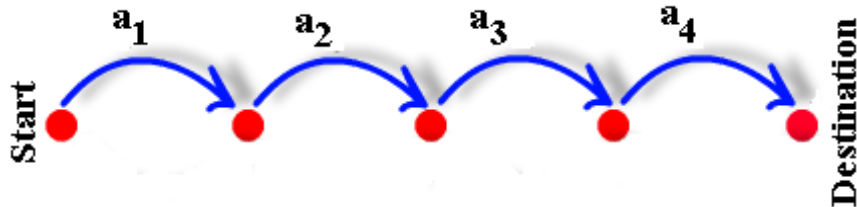

Fig. 2. First version of our device. Each arc delays the ray by the amount of time written on it. Note that this device is not complete because it cannot generate all possible subsets of A.

However, this is not enough because we also need a mechanism for skipping an arc. Only in this way we may generate all possible subsets of A.

A possible way for achieving this is to add an extra arc (of length 0) between any pair of consecutive nodes. Such device is depicted in Figure 3. A light ray sent to start node will have the possibility to either traverse a given arc (from the upper part of figure) or to skip it (by traversing the arc of length 0 from the bottom of figure).
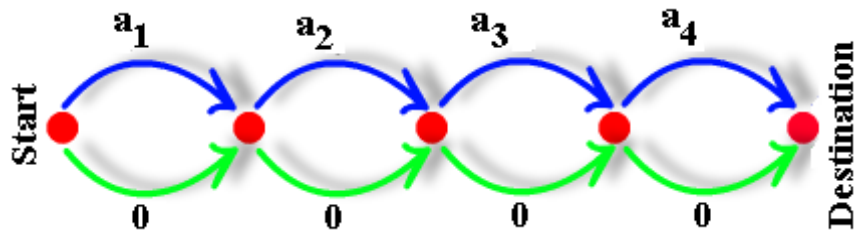

Fig. 3. Second version of our device. Each subset of A is generated, but this device cannot be implemented in practice because we cannot have cables of length 0.

In each node (but the last one) we place a beam-splitter which will split a ray into 2 subrays of smaller intensity.

The device will generate all possible subsets of A. Each subset will delay one of the ray by an amount of time equal to the sum of the lengths of the arcs in that path.

There is a problem here: even if theoretically we could have arcs of length 0, we cannot have cables of length 0 in practice. For avoiding this problem we have multiple solutions. The first one was to use very short cables (let's say of length ϵ) for arcs which are supposed to have length 0. However, there is another problem here: we could obtain for instance the sum B written as $B = a_1 + 3 * \epsilon$. Even if there is no subset of sum B, still there will be possible to have a signal at moment B due to the situation presented above.

For avoiding this situation we have added a constant k to the length of each cable. The schematic view of this device is depicted in Figure 4.
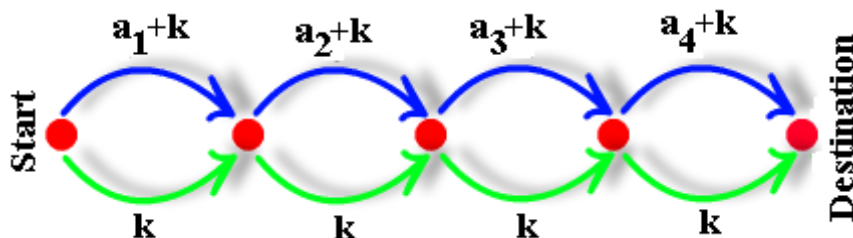

Fig. 4. A schematic representation of the device used for solving an instance with 4 numbers. On each arc we have depicted its length. There are n cables of length k and n cables of length $a_i + k$ ($1 \leq i \leq n$). This device does generate all possible subsets of A and it can be implemented in practice

We can see that each path from *Start* to *Destination* contains exactly n time value k. Thus, at the destination we will not wait anymore at moment B. Instead we will wait for a solution at moment $B + n * k$ since all subsets will have the constant $n * k$ added.

The device will generate all possible subsets of A. The good part is that we do not have to check all 2^n possible solutions. We will only have to check if there is a ray arriving at moment $B + n * k$ in the destination node. The signals generated by all other subsets are ignored and not recorded in any way.

4.4   How the system works

In the graph depicted in Figure 5 the light will enter in *Start* node. It will be divided into 2 subrays of smaller intensity. These 2 rays will arrive into the second node at moments $a_1 + k$ and k. Each of them will be divided into 2 subrays which will arrive in the 3rd node at moments $2 * k, a_1 + 2 * k, a_2 + 2 * k, a_1 + a_2 + 2 * k$. These rays will arrive at no more than 4 different moments.

In the destination node we will have 2^n rays arriving at no more than 2^n different moments. The ray arriving at moment $n * k$ means the empty set. The ray arriving at moment $a_1 + a_2 + ... + a_n + n * k$ represents the full set. If there is a ray arriving at moment $B + n * k$ means that there is a subset of A of sum B.
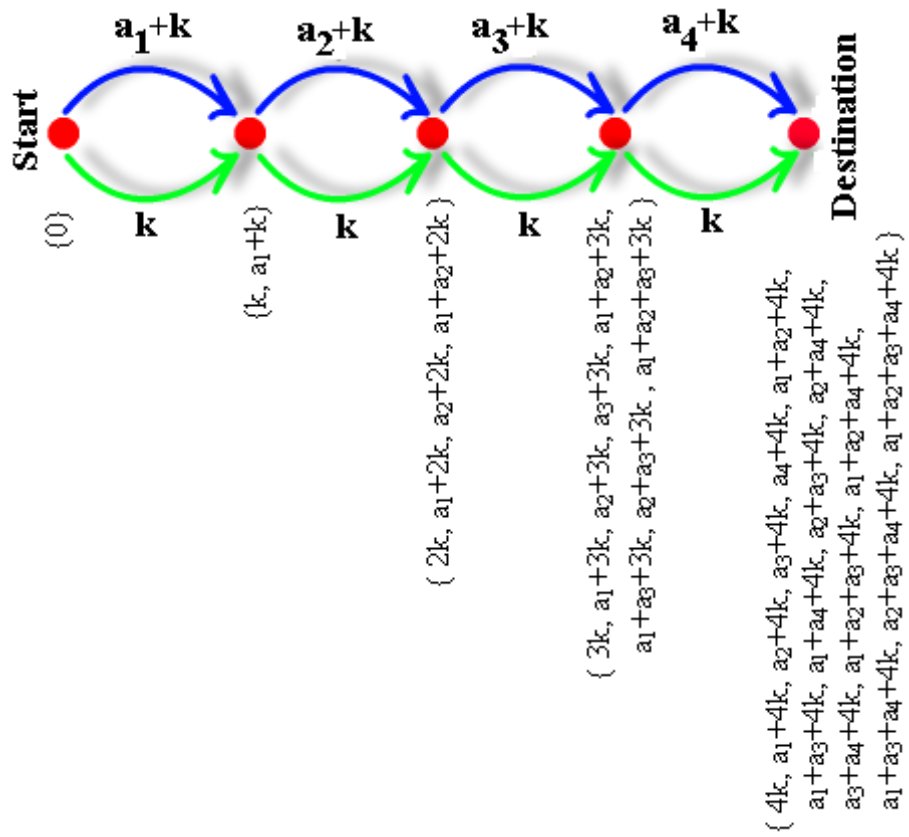
Fig. 5. The moments when different rays arrive in nodes. The moments are represented as sets because they might not be distinct

If there are 2 rays arriving at the same moment in the *Destination* it simply means that there are multiple subsets which have the same sum. This is not a problem for us because we want to answer the YES/NO decision problem (see section 3). We are not interested at this moment which is the subset generating the solution.

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 a photodiode, in fluctuations of the electric power which will be easily read by an oscilloscope.

5 Physical implementation

For implementing the proposed device we need the following components:

- a source of light (laser),
- Several beam-splitters for dividing light rays into 2 subrays. A standard beam-splitter is designed using a half-silvered mirror (see Figure 6),
- 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 lengths equals to the numbers in the given set A (plus constant k) and another set of n cables having fixed length k. These cables are used for connecting nodes.
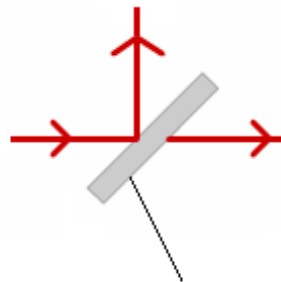
Partially reflective mirrors

Fig. 6. The way in which a ray can be split into 2 sub-rays by using a beam-splitter.

6   Complexity

The time required to build the device has $O(n*B)$ complexity. We assume that all cables are shorter than B, otherwise they cannot participate to the final solution.

Because the ray encoding the solution takes $B + n * k$ time to reach the destination node we may say that the complexity is $O(B + n)$.

The intensity of the signal decreases exponentially with the number of nodes. This is why the required power is proportional to 2^n.

7   Analysis of the proposed device

This section investigates some of the problems of the proposed device and some ways to deal with them. Section 7.1 computes the precision of solution representation. The size of the instances that can be solved by our device having a limited length for cables is computed in section 7.2. Section 7.3 shows how to handle the exponential decrease of power. Several problems that might be encountered during the physical implementation are discussed in section 7.4. Section 7.5 shows how to improve the device by reducing the speed of light.

7.1   Precision

A problem is that we cannot measure the moment $B + n * k$ exactly. We can do this measurement only with a given precision which depends on the tools involved in the experiments. Actually it will depend on the 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 if a signal arrives at the destination in the interval $[B + n * k - 10^{-12}, B + n * k + 10^{-12}]$ we cannot be perfectly sure that we have a correct subset or another one which does not have the wanted property. This problem can be avoided if all cables are long enough. In what follows we will try to compute the length of the cables.

We know that the speed of light is $3 \cdot 10^8 m/s$. Based on that 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 and it was obtained from equation:

$$distance = speed * time$$

This value is the minimal delay that should be introduced by an arc. More than that, all lengths must be integer multiples of 0.0003. We cannot allow to have cables whose lengths can be written as $p * 0.0003 + q$, where p is an integer and q is a positive real number less than 0.0003 because by combining this kind of numbers we can have a signal in the above mentioned interval and that signal does not encode a subset whose sum is the expected one.

Once we have the length for the minimal delay it is quite easy to compute the length of the other cables that are used in order to induce a certain delay. First of all we have to multiply / divide all given numbers with such factor that the less significant digit (greater than 0) to be on the first position before the decimal place. For instance if we have the set $A = \{0.001, 4\}$ we will multiply both numbers by 1000. If we have the set $A = \{100, 2000\}$ we have to divide both numbers by 100. After this operation we will multiply the obtained numbers by 0.0003 factor.

This will ensure that if a signal will arrive in the interval $[B + n * k - 10^{-12}, B + n * k + 10^{-12}]$ we can be sure that it encodes the sum $B + n * k$.

7.2   Problem size

We are also interested to find the size of the instances that can be solved by our device. Regarding the cardinal of A we cannot make too many approximations because it actually depends on the available power and on the sensitivity of the measurement tools.

However, having available a limited length (lets say 3 kilometers) for each cable, we can compute the maximal value for the numbers that can appear in A. We know that each number is less or equal to B. This is why we want to see how large B can be.

Without reducing generality we may assume that all numbers are positive integers. We know that the shortest delay possible is 0.0003 meters (see section 7.1). Having a cable of 3 kilometers we may encode numbers less than 10^7.

Longer cables may also be available. Take for instance the optical cables linking the cities in a given country. We may easily find cables having 300 km. In this case we may work with numbers smaller than 10^9. This is a little bit smaller than the largest integer value represented over 32 bits.

7.3   Power decrease

Beam splitters are used in our approach for dividing a ray in two 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 (destination node is not counted because there is no split there), each signal is divided (within each node) into 2 signals. Roughly speaking, the intensity of the signal will decrease 2^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 can 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.

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 Hamiltonian Path Problem instances of 200 cities using DNA computers [14].

7.4 Technical difficulties

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 if there was a fluctuation in the intensity at moment $B + n * k$,
- Finding a high precision oscilloscope. This is an essential step for measuring the moment $B + n * k$ with high precision (see section 7.1).

7.5 Improving the device

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.

8 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 subset-sum problem.

Further work directions will be focused on:

- implementing the proposed device,
- cutting new cables each time when a new instance has to be solved is extremely inefficient. This is why finding a simple way to reuse the previously utilized cables is a priority for our system,
- automate the entire process,
- Our device cannot find the set of numbers representing the solution. It can only say if there is a subset or not. If there are multiple subsets we cannot distinguish them. However, the subset sum 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.

References

1. Aaronson S (2005), NP-complete problems and physical reality, ACM SIGACT News Complexity Theory Column, March. ECCC TR05-026, quant-ph/0502072
2. Adleman L (1994) Molecular computation of solutions to combinatorial problems. Science 266:1021-1024
3. Agrawal GP (2002) Fiber-optic communication systems. Wiley-Interscience; 3rd edition
4. Bajcsy M, Zibrov AS, Lukin MD (2003) Stationary pulses of light in an atomic medium. Nature 426:638-641.
5. Bennett C, Bernstein E, Brassard G and Vazirani U (1997) Strengths and weaknesses of quantum computing, SIAM J. Comput., 26(5):1510-1523
6. Bringsjord S, Taylor J (2004) P=NP, cs.CC/0406056
7. Cormen TH, Leiserson CE, Rivest RR (1990) Introduction to algorithms. MIT Press
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. Gilmore PC, Gomory RE (1965) Multistage Cutting Stock Problems of Two and More Dimensions, Operations Research, Vol. 13, No. 1, pp. 94-120
13. Goodman JW (1982) Architectural development of optical data processing systems. Aust. J. Electr. Electron. Eng. 2:139-149

14

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. Kieu T.D (2003) Quantum algorithm for Hilbert's tenth problem, Intl. Journal of Theoretical Physics, 42:1461-1478

17. Lenslet website (2005), 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. 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

20. 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

21. 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

22. Oltean M (2007) Solving the Hamiltonian path problem with a light-based computer, Natural Computing, Springer-Verlag, (accepted)

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. Shor P (1997) Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer. SIAM J. Comput., 26(5):1484-1509, 1997

29. Vergis A., Steiglitz K., and Dickinson B (1986) The complexity of analog computation., Mathematics and Computers in Simulation, 28:91-113

30. Woods D, Naughton TJ (2005) An optical model of computation. Theoretical Computer Science, 334 (1-3):227-258

31. Optical Character Recognition @ Wikipedia (2006), $http://en.wikipedia.org/wiki/Optical\_character\_recognition$

32. Thoughts on the Subset Sum Problem (P vs. NP) (2006) http://idrone.net/2006/06/11/thoughts-on-the-subset-sum-problem-p-vs-np/