The University of Bucharest

              Faculty of Mathematics and Informatics

Improvement of the Approximation of the Border Length

                    in DNA Microarray Synthesis

Scientific coordinator:

Assoc. Prof.Dr. Alexandru Popa

Graduate:

Trăilă Alexandra Cristiana

Content

|  |  |
| --- | --- |
| Abreviations.................................................................................................... | 4 |
| 1. Introduction............................................................................................... | 5 |
| 2. Microarrays and BMP............................................................................... | 8 |
| 2.1 Microarray presentation .......................................................................... | 8 |
| 2.2 BMP ........................................................................................................ | 12 |
| 2.3 Formal definition of the BMP problem ................................................... | 17 |
| 2.4 The lower limits for BMP problem ......................................................... | 19 |
| 2.5 Synchronous and asynchronous incorporation for the lower limit ......... | 19 |
| 3. Background ................................................................................................. | 21 |
| 3.1 Graph Theory and Trees ......................................................................... | 21 |
| 3.2 NP-complexity .......................................................................... | 25 |
| 3.2.1 NP Class ............................................................................................ | 26 |
| 3.2.2 Difference between NP and P classes ............................................... | 27 |
| 3.2.3 Examples of NP problems ................................................................. | 28 |
| 3.3 Approximation of algorithms complexity ............................................... | 30 |
| 3.4 Formal problems ..................................................................................... | 33 |
| 4. Algorithms for BMP and AMP................................................................ | 39 |
| 4.1 The approximation algorithm for BMP .................................................. | 39 |
| 4.2 BMP approximation algorithm - case of a 1D array ............................... | 41 |
| 4.3 The approximation algorithm for AMP .................................................. | 46 |
| 5. Implementation and testing of the algorithm ........................................ | 47 |
| 5.1 Description of the BMP algorithm.......................................................... | 47 |
| 5.2 Description of the BMP algorithm implementation ................................ | 48 |
| 5.2.1 LCS distance ..................................................................................... | 49 |
| 5.2.2 Travelling Salesman Problem (TSP) ................................................ | 49 |
| 5.2.3 Partial Tree of Minimum Cost (MST) .............................................. | 51 |
| 5.2.4 Getting an Euler tour ......................................................................... | 53 |
| 5.2.5 Minimum Routing Cost Tree (MRCT) ................................ | 53 |
| 5.2.6 Algorithm testing .............................................................................. | 56 |
| 6. Final observations ..................................................................................... | 56 |
| References ....................................................................................................... | 58 |

Abreviations

BMPBorder Minimizing Problem

ADN deoxyribonucleic acid

ARN ribonucleic acid

A adenine

C cytosine

G guanine

T thymine

PCR chain polymerization reaction

1 Introduction

I chose to present "DNA microarrays" because it is an important tool in the study of DNA sequences, which are the key in detecting several diseases. Microarray DNA technologies can detect, through genes study, diseases such as lung cancer, breast cancer [39], prostate cancer [40], multiple types of tumors [43], acute lung lymphoblastic leukemia [38], colon cancer [42] ], the induction of apoptosis[44] and the way the body responds to medication [47].

Recent studies show that these can be used in order to discover single nucleotide polymorphisms of the genome [48], aberrations in methylation patterns [49], changes in the copies number of the gene [50], alternative alteration of RNA, as well as pathogen detection [ 51].

The microarrays consist of a DNA chip which is a plastic or glass plate / plate containing cDNA or single-stranded DNA in the small parts out of which it is composed. These small parts are represented in the form of a 4th order square matrix, each of which is divided into a 6th order square matrix; in the theoretical study we will consider it as a 2nd order square matrix. [31]

With their help we can check the expression of different tissues at different times. The actual process is done as follows: take a cDNA and put in the microarray to see if it hybridizes.The hybridization process is carried out by using different samples, each having a certain color; then this is placed in an analyzer and, by means of various laser beams we detect fluorescence and obtain results in the tissue expression. This is possible because all tissues in the human body have the same number of chromosomes and the same type of genes, the only difference being the way these genes manifest themselves together with the tissue. [52]

The microarrays are synthesized using two elements: with the blocking agent and the masks. The blocking agent refers to a chemical agent which, added into a space within the microarray, does not allow the addition of a nucleotide sequence in that space. This provides the ability to build within any space in the microarray a different sequence of nucleotides. We can remove the blocking agent by means of laser. The mask will cover the entire microarray, but it will have some loose spaces so that when we add a nucleotide, this one binds to the free spaces, and no addition is possible to those that are covered by the mask [62].

The microarray synthesis procedure consists in the adding the blocking agent inot all the fields of the microarray, and then the mask is placed, which will have some free spaces. Nucleotides A, T, G, and C will be added in turn to the available spaces, the unmasked ones, but they will also contain the blocking agent next to the nucleotides, and for the addition process to take place, the laser will be used to make eliminate the blocking agent. Every time we add a nucleotide, we have a new mask.

There are two types of synthesis: synchronous synthesis and asynchronous synthesis. For asynchronous synthesis there is no restriction, but for the synchronous one, the nucleotide deposition should be performed in the i-position of probes for a certain i. In this paper, we focus on asynchronous synthesis, which is harder to optimize but much more flexible than the synchronous synthesis [29] [30] [31].

What the microarray synthesis brings new, along with the existing DNA, PCR and Southern and Northern blot technologies, is that it measures the levels of expression of thousands of genes in the same experiment [52].

There are two steps in the synthesis of DNA microarrays: probe placement and probe inclusion [31]. Used in the synthesis, the probe placement refers to the placement of each probe in a single location in the microarray, and the integrity of the probe is the sequence of steps performed with and without the mask.

In this paper we will outline algorithms that solve the problem of minimizing borders, called BMP, that occurs when we place and embed probes. The synthesis is accomplished by using a chemical process in which light plays a role, but in this case it also appears the unintended illumination that might affect the results. In order to control this undesirable phenomenon, the boundary length should be measured, proving the amount of unintended lighting, and the BMP needs to find a placement destination and probe integration so that the border length is minimized. The BMP problem has been shown to be NP-hard. This paper presents the algorithm implemented for BMP that has an approximation of complexity O(), and if the placement destination and integration of probes is known, then the problem is O(), where n is the number of probes to be synthesized [29] [31].

In the following sections there are presented the necessary notions which are part of BMP and the algorithms that characterize this problem. In section 2, there is a presentation of the context of microarrays and of the BMP problem. Section 3 provides information about the data structures and algorithms representing the basis of the problem solving, as well as a wider display of NP and P complexity classes, and a characterization of algorithms given by the length of time an algorithm needs to run. Section 4 describes the BMP algorithms. In Section 5, there are presented the ways of implementation and some tests for a clearer presentation of the algorithm efficiency. Finally, the conclusions drawn from the study of this problem as well as an idea for improving the algorithm will be presented.

2. Microarrays and BMP

2.1 Microarray presentation

DNA microarrays or DNA chips are a powerful tool for studying DNA sequences. These are small -sized glass or or plastic surfaces containing thousands of spots. Each of these places contains a DNA sequence, which shall be further on referred to under the name of sample [1] [2] [3].The DNA sequence is represented either by a complete copy of genomic DNA or by a short sequence of oligonucleotides corresponding to a gene [4]. They can identify diseases such as Parkinson's disease, cancer or Avian Influenza (H5N1) [37].

The concept and methodology of microarrays were first introduced by Tse Wen Chang in 1983 for antibodies microarray, a microarray for proteins that detect their expression in plasma and cell, tissue and serum, and became increasingly popular after 1995 [5]. Thus, microarray DNA technology has become one of the most sophisticated and most used technologies for studying DNA [6]. There are several types of microarrays, besides DNA, such as: MMChips, Protein microarrays, Tissue microarrays, Peptide microarrays and others [7].

The way the microarrays function is the hybridization of two DNA strands, i.e. the property of complementary nucleic acid sequences that specifically bind to one another by forming hydrogen bonds between complementary base nucleotide pairs: A with T or T with A and C with G or G with C. DNA hybridization means that a single DNA strand, also called ssDNA, is attached to another ssDNA coming from another source. Each place within the microarray contains sequences from certain genes represented by ssDNA. Each sequence is put in a specific way and place so that each position in the chip corresponds to a particular DNA sequence [8] [9].

To make it clearer how microarrays are used, we will exemplify how they work on two samples obtained from an uninfected plant and one infected by the virus. From both plants, the mRNA is extracted, which is a RNA molecule that has the role to copy the genetic information of a DNA strand. This, the mRNA, is transformed into cDNA by means of an enzyme called reverse transcription. Each of the two types of samples, uninfected and infected, is labeled with a specific color, e.g. green fot the uninfected, and red for the infected one. At this point, the two cDNAs obtained are combined and applied on the microarray.

Each cDNA binds to the complementary DNA sequence and will not bind to another one; this is called the hybridization step. After this step, any cDNA that has not formed a bond is removed. Further on, this DNA chip will be placed in a device which electronically capture data, scanning the chip with a laser. This one activates the fluorescent color in the cDNA samples. At this point, there is a computer that has the role of processing the obtained information, calculating the ratio of green and red in each place. This will indicate which genes are active in the samples obtained from uninfected and infected plants [8] [9].

Applications include genotyping, expression analysis, and sequencing. New ways have been sought to improve the flexibility and widen the use of microarrays. One of these innovations concerns the introduction of mirrors to direct light [10].

Prior to finding DNA chips, many of the standard methods for finding pathogenic characteristics, such as drug resistance and virulence identification were performed on laboratory cultures - a difficult process from several points of view, including the long duration due to their cumbersome growth. Thus, through this laboratory method, the results that provide information about phenotypes can no longer be clinically relevant. Since many molecular tests based on DNA sequencing are used in microbiological diagnosis, the microarrays are therefore very useful, providing more precise and effective results regarding the genotype and the manner in which certain phenotypic characteristics, such as drug resistance, are expressed.In order to identify several bacterial pathogens and drug resistance markers, using a standard chip could be a cost-effective diagnostic tool. To achieve this goal, an universal sequencing chip is under progress. It will provide valuable information regarding diagnostic applications [16].

An example to prove that microarrays are a diagnostic tool is the detection of viral pathogens. A way to identify virulence markers are low density oligonucleotides that define either short sequences of interest or individual mutations. The DNA chips have the ability to find the genotype of viral pathogens and can be useful in determining viral-site transmission pathways. The most common actions performed with DNA microarrays are the study of cells, viruses and bacteria. To perform the microarray analysis on bacteria, the mRNA must be extracted quickly because its time is short in many organisms [10-15]

A continuous observing of the levels of expression of certain genes in pathogens may be important for the diagnosis of particularly virulent strains or to compare the response to different drugs [17]. This may be beneficial for organisms that initially do not respond to therapy, in order to find an efficient medical treatment. The new studies show that microarray analysis can be used to study the expression of the human cytomegalovirus and herpes simplex virus gene [18] [19].

Using the DNA microarrays, the interaction between a host and an infectious agent such as microbes, viruses, parasites, prions may be studied as well.

The way a host responds to an invading pathogen involves several mechanisms which can be now studied in detail, by means of DNA chips. This can also help to identify people most susceptible to a particular infection and prognostic marker to find out where it started.This result might help to identify the populations that have a certain predisposition to a particular infection and to take timely action. If a person is suffering from a particular infection, it is possible to identify the most effective way to treat it [20] [21] [22].

It is already known that comparing gene expression patterns is a means of classifying tumors in cancer studies in humans [23] [24]. It has been found that gene expression patterns induced in human primary monocytes infected by two strains close to Ebola, Zair and Reston are significantly different.Thus, host expression patterns could be used in the diagnosis of infectious diseases and as an indicator of the disease progression. Microarray technology has managed to differentiate between the two strains of Ebola [25].As thousands of genes are being studied simultaneously, the development of algorithms is needed to analyze large amounts of produced data. In this case, the term "data-mining" is used, which includes the classification and aggregation of the information extracted from the analysis of all data provided by the microarray.The aggregation of the extracted information deals with the organization of data in groups of genes or samples with similar patterns that are characteristic of the group. Classification is a learning process from the examples. Therefore, with a set of pre-classified examples, a rule to assign new samples to one of the previous classes will be found. [26] [27].

2.2 BMP

Following the presentation in the previous subchapter about the context of DNA microarrays, we can now present the problem this paper is studying, the Problem of Minimizing the Border. The English name being Border Minimization Problem, we will continue to use BMP abbreviation. Microarray synthesis is accomplished through a process called large-scale immobilized polymer synthesis synthesis (VLSIPS). To add a DNA sequence to each location in the chip, called synthesis of microarrays, two components are used: a blocking agent and masks. The blocking agent has the role of not letting a nucleotide form a link to the nucleotide sequence from a particular place. The mask will cover the places in the microarray, but it will have some free spaces as well [28].

The first step in this process will be to put a blocking agent in each place. The second step is to put a mask that will have some gaps in certain places in the microarray. Now, if we want to add a nucleotide (represented by letters A, C, G, T), we exclude the blocking agents with a laser, and this can happen only in the places where the mask has gaps. Further on, that nucleotide will be placed into the gaps given by the mask. Nucleotides are added in turn and not all at once, and each contains a blocking agent. Thus, if a new nucleotide is to be added, the synthesis process steps will resume starting from putting a new mask and using the laser to exclude the blocking agent from the gaps in the mask. This laser is represented by a ultraviolet illumination.Using masks, at each location in the microarray, different nucleotide sequences can be obtained by placing them before illumination. Areas that do not receive illumination (the masked ones) will not be coupled to the new nucleotide layer. Thus, by adding multiple masks and nucleotide sequences, a DNA chip is obtained that has many different built-in probes, one in each place. The nucleotide sequence and masks required to construct all the samples in a given place is called a deposition sequence.

Reprotection

Mask

UV source

Nucleotide layer

Unprotected areas

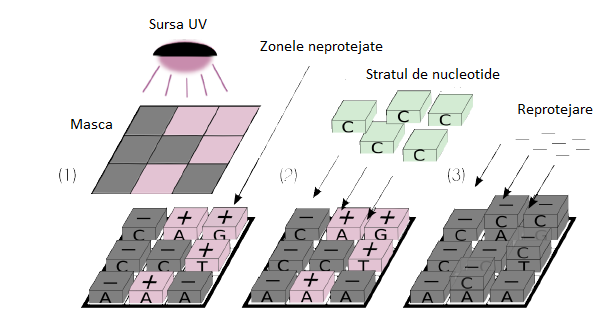


Fig. 1: The process of adding nucleotides. (1) Ultraviolet illumination shows areas where the nucleotides will be added (areas marked with a plus). (2) Adding nucleotides. (3) Reprotection of all areas

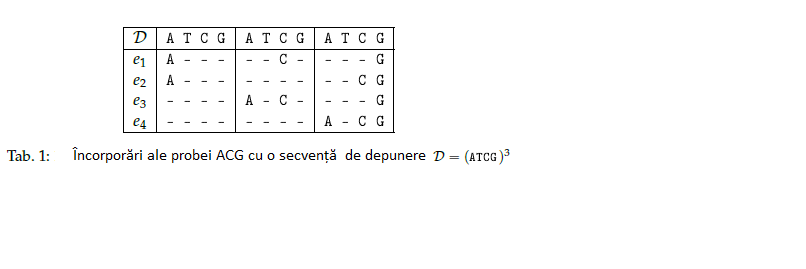
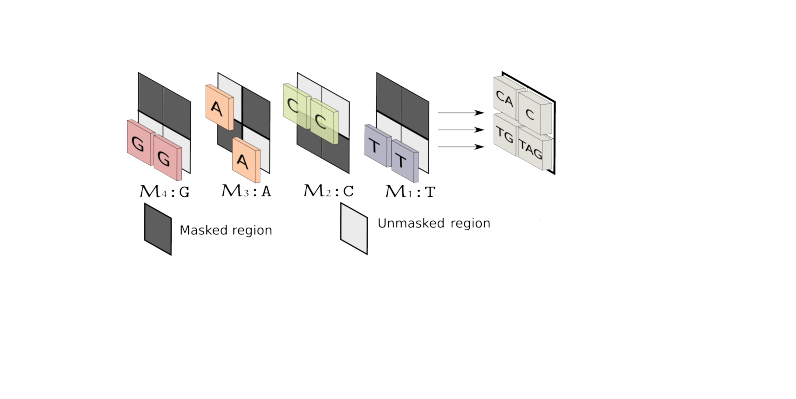
In this process of deposition of nucleotide sequences in the microarray, two problems may arise. The first problem is called probe placement and refers to choosing the position of each probe in a given location. Another problem is choosing the nucleotide sequence and masks that construct the required probe in the selected place, this is called probing embedding problem. The two might sum up the question of "where to put the nucleotide sequence", respectively "how to put it" using masks [3].

Table 1: Embedding of the ACG sample with a deposition sequence

We get the deposition sequence D = TCAG. Then a possible embedding for the placement of the probe set {CA, C, TG, TAG} is shown in Figure 1.2. In this case, 4 layers of masks will be required. Each mask is marked by Mi, where i indicates the order of the masks.

Depending on the size of the deposition sequence, there are several different embeddings for each sample. Table 1 illustrates some of the possible embeddings for the ACG probe and the deposition sequence D= (where ε1, ε2, ε3, ε4 are embeddings). The "-" symbol means a probe masking during the microarray synthesis process. Removing spaces from these embeddings will result in the original probe.

Fig. 1.2: Synthesis of a matrix 2 × 2. Four (4) masks, M1, M2, M3, M4, are applied. Boundaries are represented by bold black lines. For M1, M2, M4 the length of the boundary is 2. For M4 the length of the boundary is 4. In total, the length of the border will be 10.

There are two types of embedding: synchronous embedding and asynchronous embedding. In synchronous synthesis, each deposition nucleotide may be deposited only at the i position of the probe for a particular i. This involves the use of a periodic deposition sequence ,where k is the length of the probe. Thus, each nucleotide at the position i in the deposition sequence shall be incorporated during cycle i of ACGT. An example of synchronous embedding is represented in Table 1 by embedding ε1.Asynchronous embedding refers to any possible embedding, and it is not restricted by any rule. Except for ε1, the other embeddings remaining in Table 1 are asynchronous. Asynchronous synthesis may ease the synthesis process, but it is more difficult to optimize it [3] [31].

Because of the unintended illumination, the masked areas can be affected by the addition of unwanted nucleotides in that place, and thus the experimental results are compromised. This happens because the illumination may also bring along the diffraction of light, internal reflection and scattering. The unintentional illumination is given by the number of boundaries between masked and unmasked areas, this being called the border length. In Figure 1, the border length for M1, M2, M4 este 2, while for M3is 4.

Although the unintended illumination can be reduced by technologies used for synthesis, the problem persists [3]. To solve this problem, we can use the fact that we can choose where to put the probe and the way we choose to put it. The Border Minimization Problem (BMP) tends to minimize the total boundary length between masked and unmasked areas. This is done by optimally finding the placement and embedding of the probe. Because with BMP any placement and sample embedding can be chosen, the areas that are unintentionally illuminated are reduced. Because of the exponential number of possible probe insertions, the problem is NP-hard. An issue that restricts the placement of the sample to a particular place is called P-BMP. This is a BMP subproblem, but it only deals with finding an optimal embedding. The P-BMP problem can be reduced to other NP-hard problems, such as the weighted minimum sequence alignment (WMSA) and the minimum routing cost tree (MRCT). This helps to study the complexity of the BMP with the complexities already known from the aforementioned problems [29] [30].

The BMP issue was officially presented by Hannenhalli and his collaborators [32]. The most common approach to solve the problem of placement is the one that uses the Traveling Salesman Problem (TSP) [3].

Carvalho and Rahmann use heuristics based on some greedy techniques [28] [33] [34]. An important contribution was provided by Kahng and his collaborators who presented comparisons between BMP algorithms and achieved their own heuristics. They obtained lower boundaries for both cases, both synchronous and asynchronous [35] [36].

Li and his collaborators offered the first algorithm with superior boundaries to approximate the error, this being O(n). Popa and his collaborators have improved this limit to O(n1/4 n) by a placement that uses the LCS distance in a trees distribution using a probabilistic embedding of the metrics [29] [31].

2.3 Formal definition of the BMP problem

In this subchapter we will provide a formal description of the BMP problem described in Li's article. A matrix is given of size, where is considered to be an integer for simplicity. We have a samples set P={p} with tha appropriate lengths l1, l2, …, ln. Two areas within the matrix, c1 și c2, are considered to be neighbors if |x1– x2|+ |y1– y2| = 1.All neighbors of region c are marked by N (c). A placement destination for P is a bijective function φ, assigning to each sample pi a single location in the matrix.

Having a deposition sequence D, an embedding for the sample set of P is represented by a set of sequences ε, where ε = {ε1, ε2, …, εn} so that:

1. Each sequence εi has the length |D|
2. εi[k] = D[k] or εi[k] = { - } for all I belonging to the interval [0,n]
3. by eliminating all spaces “-” out of εi, we get pi.

The Hamming distance between two sequences of equal length is defined as the minimum number of characters that need to be modified so that one of the sequences converts to the other. The boundary between two neighboring samples, pi și pj, is measured with the Hamming distance for the corresponding embeddings, εi and εj respectively, and is denoted by the borderε(pi, pj).

Definition 1: BMP aims to find a destination φ and an embedding so that the border length BL (φ, ε) is minimized.

The length of the border can be defined in two ways. The first of these is defined as the sum of all the borders of all sample pairs in the matrix:

BL(φ, ε) = borderε(pi, pj)

pi, pj  
 ϕ(pi)∈ϕ(pj)

The other definition is based on the masks by means of which we form the nucleotide sequences in each place. Given the set M = {M1, M2, ..., Mn}, each having the border length BL (Mi), then the formula that calculates this length shall be as follows:

n

BL(φ, ε) = BL(Mi)

i=1

The BMP has been for a long time considered to be a NP-hard problem, but without any formal demonstration of it. Li had shown that in case 1D, P-BMP is solvable in polynomial time. The case 1D refers to a single-row matrix. [33] [34].

Prior to Popa's article, there was no stronger result demonstrating that the BMP problem is NP-hard in any size, and for the P-BMP problem it has been demonstrated that this is NP-hard larger in size than 1D. It has already been shown that P-BMP is reducible to other NP-hard problems, having predetermined locations where each probe shall be placed. Since the results obtained so far do not indicate the existence of polynomial solutions of the problem, approximation algorithms are still sought [29] [31].

**2.4 Lower borders for BMP problem**

Kahng and his collaborators have shown the limits for the total length of the border. Lower limits exist for both synchronous and asynchronous incorporation [35Kahng și colaboratorii săi au arătat care sunt limitele pentru lungimea totală a frontierei. Limitele inferioare există atât pentru cazul încorporării sincrone , cât și pentru cea asincronă [35].

**2.5 Synchronous and asynchronous incorporation for the lower border**

The lower limit for the total length of the border for the synchronous embedding of the set of probes P into a matrix m × m is given by the Hamming distance according to the rule:

The sum of the four probes closest to each sample.

For an already known placement, for a set of samples P placed in a matrix m × m, the lower limit for the total border length for asynchronous embedding is given by the LCS distance as follows:

Sum of adjacent samples of each probe.

3 Background

In this section we present some notions about graph theory, NP-complexity with its subsections, and the approximation of the complexity of algorithms, these being necessary for understanding the algorithms presented in sections 4 and 5. The following notions have the theoretical background of Cormen's book "Introduction to Algorithms" and the book "Fundamentals of Programming" by D.Lica [52] [54] [63].

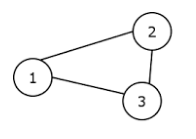


Fig. 2.1

3.1 Graph Theory and Trees

This section presents elements of the graph theory which are the very base of this work. The graph theory is a vast subject, with many applications in fields such as the economy; some of the problems in this field solved by means of the graphs are: the length of the road between two localities, the duration of a route, the quantity transported on that route and many others. This proves that graphs are a useful and powerful tool. Further on, there are presented several notions that go through some of the graph theory elements [53].

Definition 1 (Graph) : It is called a graph an ordered pair of sets, denoted G = (V, E), where V is a finite set and non void of elements called nodes or peaks, and E is a set of pairs (ordered or unordered) of V elements called edges (if there are pairs) or arcs (if there are ordered pairs). In the first case, the graph is called non-oriented, otherwise it is an oriented one.

To refer to an edge of E, I will use the notation [x, y] that is the same as the edge [y, x].In the theory of graphs, the following notions are very common:

- the extremities of an edge

• given the edge m=[x,y],the nodes x and y are called its extremities;

- vârfuri adiacente

if in a graph there is the edge m = [x, y], it is said that nodes x and y are adjacent;

- incidence

• if m1 and m2 are two edges of the same graph, they are called incidents if they have a common end

Definition 2 (degree): The degree (or valence) of a node in a graph is the number of incident edges with the node. This is denoted as deg (v), and the maximum degree and minimum degree of a graph G is symbolized by Δ (G), respectively by δ (G). The maximum degree (minimum degree) means the highest (lowest) degree of a node.

Example: In the graph in Figure 2.1 the maximum degree is 2, and the minimum degree is 2 as well.

Definition 3 (Regular graph): In a regular graph, all degrees are the same, so we can talk about the degree of the graph.

Example: Figure 2.1 shows a regular graph.

Definition 4 (Chain): A chain is a sequence of nodes of a non-oriented graph G = (V, E), with the property that any two consecutive nodes are adjacent:

w1,w2,w3 ..., wp with the characteristic that (wi, wi+1)belong to E for 1≤i<p.

A chain in a graph can be of three kinds:

- the single chain ⇔chain containing only distinct edges;

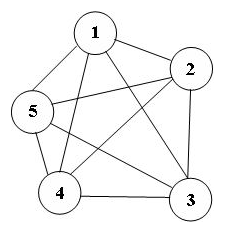
-the composed chain ⇔the chain that does not consist only of distinct edges;

-the elementary chain ⇔the chain that contains only distinct nodes.

Definition 4 (path): It is called a path in a chart a succession of adjacent and distinct edges connecting two peaks in the graph (called the ends of the path).

Definition 5 (Complete graph): Non-oriented graph G = (V, E) where there is an edge between any two nodes. The number of edges of a complete graph is |V|\*|V-1|/2.

The following figure shows an example of a complete graph:

Fig. 3.1

Definition 6 (Cycle): A cycle is a chain where the first node coincides with the last one. The cycle is elementary if it is formed only from distinct nodes except for the first and last. The minimum length of a cycle is 3.

Definition 7 (Subgraph): A subgraph of G = (V, E) is a graph G '= (V', E ') where V'⊆V and V' contains all edges / arcs in V '. In other words, G 'is a subgraph of G, if it is identical, or it is obtained by suppressing some nodes together with the edges / arcs incident to them.

Definition 8 (Partial graph): A graph G '= (V, E') is a partial graph of G = (V, E) if E'⊆E . I.e., G 'is a partial graph of G, if it is identical, or it is obtained by suppressing some edges (or arcs) of G.

Definition 9 (Related Chart): Non-oriented graph G = (V, E) where, for any pair of nodes (v, w), there is a chain that joins them.

Definition 10 (Tree): A tree is a non-oriented, related and acyclic graph (without cycles)

Definition 11 (Root Tree): A root tree is a non-oriented, related non-cycling graph where one of the nodes is designated as root. The nodes can be placed on the levels starting with the root, which is placed on the first level.

The concept of descendant is as follows: In a root tree, y is the descendant of node x, if it is located on a higher level (as a sequence number) than the level of x, and there is a chain that joins them and does not pass through the root.

Definition 12 (Peak Level): In an arborescence, the peak level x is the path length from the root to x.

Definition 13 (Weighted tree): It is called a weighted tree if each edge is associated with a numerical value called "cost".

Definition 14 (Partial Tree): Given a related non-oriented graph, a partial tree graph is a partial graph with the characteristic of being a tree. I.e., a partial tree is a tree obtained by removing some edges from the graph.

3.2 NP-complexity

This section shows what NP-complexity means. The problem we are trying to solve, BMP, is NP-hard. Below we shall give some ideas on this subject expressed through some definitions and examples. More details on this subject can be found in books such as: "Introduction to the Theory of Computation" written by Michael Sipser and another reference book, but older than the first, is "Introduction to Automata Theory, Languages, and Computation "written by John Hopcroft and Jeffrey Ullman. The following statements are based on Cormen's book, "Introduction to Algorithms".

3.2.1 NP Class

It is considered that we have a set of P programs, n processors, a t (p) execution time for each program p belonging to P, and a program settlement term D. It is required to find a processor planning for the P programs so that any program might be executed within D. This problem is NP-completeness. The NP class was defined by Stephen Cook in 1971 in his book, "The Complexity of Theorem Proving Procedures".

Definition 15 (NP class): The NP completeness class (non-deterministic polynomial) comprises decisional problems that are executed in the worst case in polynomial time by a non-deterministic Turing machine.

It can be said that the NP class is the class of problems that can be solved by non-deterministic algorithms in polynomial time.

Definition 16 (NP-hard class): A H decision-making problem is NP-hard, when for each L problem in NP there is a polynomial time reduction of L to H.

This definition means that any problem in the NP-hard class should be at least as difficult as the worst NP class problems [63] [64] [65].

Definition 17 (NP-completeness class): In the complexity theory, a decision-making problem is NP-complete when it is in both NP and NP-hard.

An informal definition is that the NP-completeness class is the decision-making class that contains the most difficult NP problems. Each NP-complete problem must be in NP [63] [65] [66].

3.2.2 The difference between NP and P classes

A problem that is part of the complexity class P is stated as follows: there are N localities and M roads between localities, each of these roads having a certain cost. It is required to find the shortest route between two specific localities. This problem can be solved using Dijkstra's algorithm that has a complexity of *O(n2)*  [54].

Since this is the upper limit that the algorithm can have, the conclusion is that there may be two localities so that the algorithm can be solved in a less complexity, and so it can be said that this problem belongs to the P complexity class. Further on, we present the definition of P complexity classes [54] [63].

Definition 17 (P complexity class): The P complexity class contains the decisional problems that are executed in the worst case in polynomial time by a deterministic Turing machine [63].

This definition presented in an informal manner, based on the example presented at the beginning of this section, refers to the fact that the most unfavorable case of solving the problem of finding a path between two localities has the complexity O (n2) [54] [63] [64] [65].

After presenting the two notions, both the NP- complexity class and the P complexity class, we can say that the NP problems can be quickly checked and those in P can be quickly solved.

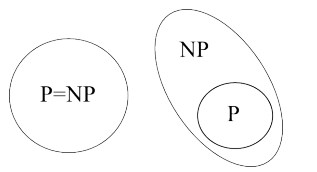
One can notice that P⊆NP, but the question is whether P⊂NP or whether P=NP. This is one of the most familiar questions. It is of major importance, because finding an answer may help you may find the solution to many problems as well.

Most scientists consider P ≠ NP. The P = NP response would be problematic for many people, for example for banks, because their encryption system is based on the inequality between P and NP, and if the opposite is proved, they would have to change the system [ 63] [54]

For the time being, only the following can be demonstrated:

Theorem 1 :If P is in NP, then there are polynomials p (n) and q (n) and a deterministic algorithm that solves P during O (p (n) 2q (n)).

Figure 3.2.2presents the illustrated hypothesis P=NP or P⊆NP.

Figure 3.2.2

3.2.3 Examples of NP problems

This section concerns some of the issues considered as parts of the NP complexity class for a better understanding of this concept.

One of the most common problems in this class is the SAT problem (the problem of satisfiability). Its statement is as follows: given a formula F from the propositional calculation in the normal conjunctive form in which the variables of {x0, ..., xn-1} appear. The requirement is to see if there is an assignment of variables for which F is satisfied [63] [54] [66] [65].

Theorem 2 (Steven Cook):One of the most common problems in this class is the SAT problem (the problem of satisfiability). Its statement has the following form: is given a formula F from the propositional calculation in the normal conjunctive form in which the variables of {x0, ..., xn-1} appear. The requirement is to see if there is an assignment of the variables for which F is satisfied [63] [54] [66] [65].

Algorithm for the problem of satisfiability:

Input: Let the formula F be in the conjunctive normal form consisting of the variables{x0,...,xn-1}.

Output: Finding the appropriate formula variables brings about either "Satisfied Problem," or "Unsatisfied Problem".

Algorithm:

1). Random assignment for variables is taken

2). With the values found in step 1), the function value F is calculated

3.) If the formula is satisfied, then it displays "Satisfied Problem" otherwise it displays "Unsatisfied Problem" and resumes step 1) for all possible assignments for variables.

This is a correct but not efficient algorithm. The complexity of this algorithm is about 2n, where n is the number of variables [54] [63] [64] [65].

There is a number of problems that can be reduced to the SAT, such as:

- We are given a map and several cities, the question is whether there is a path going through each city exactly once. This is called the Hamiltonian cycle problem [54].

- The problem of the backpack, which says that we have a backpack and some objects, each characterized by some its weight and a certain profit. The problem requires that a subset of objects shoul be found so that the sum of their profits is the maximum, and the sum of their weights does not exceed a value G; the G value indicates the maximum capacity that the backpack can support so that it does not break [54].

- Another problem which that can be reduced to SAT is the problem of coloring a two-color map [54][63].

Another problem that is NP-complete is represented by the square congruences.

Input: We get three numbers: a, b, c positive

Output: We are asked to find a number x which sould be comprised between 0 and c, 0 <x <c, and the second condition is that the remainder of x raised to square divided by b shall be a, ie x2 mod b = a.

3.3 Approximation of algorithms complexity

To solve a problem, the most frequent requirement is, in addition to a correct algorithm, also an efficient one. The term "efficient algorithm" usually means an algorithm that uses some structures to occupy as little memory as possible, but also to solve it in a short time. Generally, these two requirements, in order to have an efficient algorithm, can not be simultaneously accomplished.Depending on the problem and what result is expected, we can give up some efficient data structures to increase the speed of the algorithm, and vice versa. We can give up one in favour of the other. To solve this problem, a middle path is sought, meaning that the most appropriate combination of the data structures used and the speed of the algorithm are required [66] [63] [54].

In this section we shall describe the time complexity of. This is a notion that evaluates the effectiveness of algorithms. Here are some formal definitions.

Definition 18 (Big-Θ): We say that f (N) is Θ (g (N)) if f (N) is O (g). A representation of this notion is found in figure a) [54].

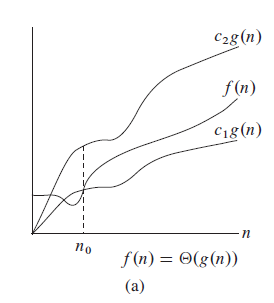
This notion asymptotically limits a bottom and top function.

Definition 20 (Big-O): We say that f (N) is O (g (N)) if there exists c and N0 so that: for any N> N0 we have f (N) < c\*g(N). This notion is used when we have only an asymptotic upper limit. A representation of this notion is found in figure b) [54

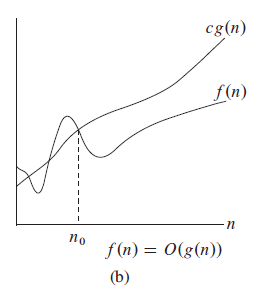
Definiție 19 (Big-Ω) : Spunem că f (N) este Ω(g(N)) dacă g(N) este O(f(N)). O reprezentare a acestei noțiuni se regăsește în figura c) [54].

Just as Big-O offers an asymptotic upper limit, Big- Ω offers an asymptotic inferior limit.

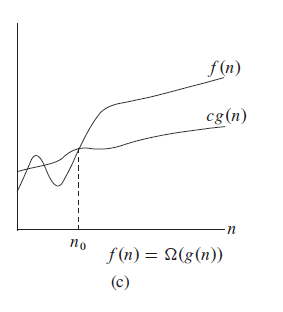
The following figures graphically describe the mentioned notions.



It can be noticed that the function f (n) is asymptotically bordered both at lower and higher level by the function g (n), multiplied by the constant c1, respectively for the upper boundary with the constant c2 [54].



The function f (n) is only bordered in the asymptotic upper part by the function g (n) multiplied by the constant c.



The function f (n) is only bordered in the asymptotic lower part by the function g (n) multiplied by the constant c.

3.4 Formal problems

In this section we present formal problems, together with the algorithms used, which are the basis for solving the BMP problem. This includes the graph and tree chapters along with various algorithms specific to these chapters.In BMP solving algorithms, appear notions such as the tree with minimal routing costs (MRCT), TSP problem, precommination common subsequences, but also the Hamming distance. We shall define hereunder these notions [29] [31] [54].

For the trees, there are several modes of representation, such as:

- adjacent matrix

- the list of "descendants"

- fathers'vector, which can be used only for trees with a designated root.

Definition 20 (Adjacent matrix):

The adjacent matrix is a matrix A where the elements a[i,j] are defined as follows::

* 1, if edge [i,j]exists with an i differentfrom j.
* 0, otherwise.

The Fathers' Vector stores for each node its parent. For the root, the corresponding element in the vector is equal to 0. In the following example we present the vector for the given tree.

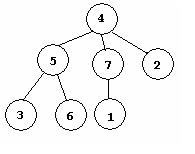


Fig. 3.4 :Fathers’ vector is T= (7, 4, 5, 0, 4, 5, 4)

Determining a partial tree can be done using a graph scroll algorithm, either in breadth or in depth.

Browsing BF graphs in breadth - Breadth First

This method is based on crossing all edges available from the current node to the adjacent unvisited nodes, which shall be visited in this manner. After this process, the explored node is removed from the queue, successively processing all the nodes reached at the top of the queue.

Thus, it can be observed that the basis of this method of browsing is the FIFO type mechanism [52] [54].

The manner this algorithm unfolds allows for the identification of the minimum length chains from the starting node to all the peaks that can be reached from the graph [52].

The BF tree formed by the edges crossed during the browsing in breadth has the property of being formed only by chains of minimum length; these chains join the starting node to all the accessible peaks [52].

An algorithm based on this breadth crossing strategy is "Lee's Algorithm" which determines the minimum chain between two peaks of a graph, the starting node being one of the two given peaks [54].

Other applications for the breadth browsing are the following:

- Checking whether a graph is related

- Determining the related components of an non-oriented graph

- Checking whether a related graph is bipartite

- Converting a necones graph into a related one

Here below, we shall present the iterative algorithm of BF.

Algorithm for a breadthbrowsing [54]

Input: Let thegraph *G=(V, E),*where V represents the nodes set and E the edges set.

Output:Display of nodes in case of breadth browsing -BF

Algorithm:

1)Put the first peak in the queue.

2)All the peaks adjacent to the node are identified and are inserted after the last peak in the queue if they have not been visited.

3)Repeat the process from step 2) for the next node in the queue.

4) The algorithm ends when the end of the queue is reached.

If N is the cardinal of V and M the cardinal of E, then we can define the complexity of the BF algorithm presented above. This is *O(M+N*) [54].

Browsing BF graphs in DF – Depth First

DF browsing works by following the LIFO mechanism. The strategy of browsing in depth of a non-oriented graph involves crossing an edge of incidence at the current peak to an adjacent unvisited peak. When all the edges of the current peak have been exploited, we return to the peak that led to the exploration of the current node.This process is repeated until all the peaks have been browsed. The peak that is removed from the stack has no edge available for crossing [52] [54].

The recursive version of the algorithm is presented below.

Algorithm of depth browsing graphs DF - Depth First [54]

1) Put the given node as visited.

2) The neighbors of this peak are searched and they must be unvisited as well.

3) Apply step 1 again for the node found in step 2).

If N is the cardinal of V and M the cardinal of E, then we can define the complexity of the DF algorithm presented above. This is*O(M+N).*

Kruskal's algorithm

We have a non-oriented graph G = (V, E) with costs associated to edges, where N is the cardinal of the set V, and M the cardinal of the set E. The Kruskal algorithm finds a partial minimum cost tree. It builds partial minimum cost arnors, starting from N disjoint trees.Each peak of the graph defines one tree at the initial moment. At each step of the algorithm, two trees will be chosen, which shall unify. Finally, the minimum cost tree will be obtained. In order to obtain a tree through the unification of two trees we should do the following: take the minimal cost edge, previously unused, and which has the extreme peaks in two disjoint trees [54] [52].

Kruskal’s algorithm [54][52]

1) Create a set of trees where each peak in the graph is part of a separate tree.

2) Build a set containing all the edges in the graph

3) As long as the set of step 2 is non-void, the following operations are performed:

* + Remove a minimum cost edge from the set from step 2)
  + If that edge connects two distinct trees, then add the edge to the forest, combining the two trees into one, i.e. ignore the edge [67].

The complexity of Kruskal's algorithm is *O (N \* M),* where N is the cardinal of the set V, and M the cardinal of the set E [54] [67].

Further on, having the notions which are the basis of the description of the BMP, we present the algorithms for BMP.

4. Algorithms for BMP

In this section, the BMP is being studied. Here are some algorithms that solve this problem. We need to find a placement destination and an embedding for the given set of probes. An approximation algorithm shall be given O(),this one being based on a BMP variant called P-BMP. We shall also discuss the case where we have a one-dimensional array and it shall be proven that it admits better results in this case. This chapter closely follows the presentation of the algorithm made by Li and his collaborators [31] [30] [29].

4.1 The approximation algorithm for BMP

Theorem 2: A variant of the BMP problem, called P-BMP (BMP problem where placement is given), may be polynomially reducible to WMSA (alignment of several weighted sequences).

The P-BMP problem shall be studied, a BMP variant with a pre-determined placement. Finding an embedding becomes the main preoccupation. The P-BMP problem can be approximated O (log2 n) giving a reduction to the problem (WMSA) for which there is an approximation algorithm O (log2n) [31] [29].

Lemma 1: There is a polynomial time reduction from P-BMP to WMSA [31].

In order to use the result for P-BMP, we need a particular placement destination , the choice of which is guided by the TSP problem on a graph. Finding a minimum TSP is NP-hard, but there is a polynomial time O (1) of approximation [54] [31].

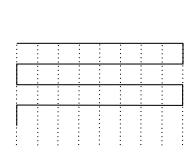
The graph built in the algorithm is a fully weighted graph with nodes representing the probes and the weight of the edge representing the distance between two nodes.

The length of the border is closely related to the common subsequence and the common supersequence between adjacent sequences in the placement site. Take any two length sequences having a length l, for example p and q. The longest, respectively shortest common subsequence of the two sequences p and q by LCS (p, q) and SCS (p, q), respectively, and the corresponding length as | LCS (p, q) |, respectively SCS (p, q). SCS (p, q) can be obtained by finding LCS (p, q) and inserting in p the characters of q that are not of LCS (p, q), keeping the order in q [29] [30][31].

Therefore, | SCS (p, q) | = 2ℓ − | LCS (p, q) |.For any embedding ε, the maximum number of common nucleotide deposits between p and q is | LCS (p, q).

The distance between p and q shall be denoted as dist (p, q), where it shall be equal to the LCS distance which shall be defined as 2(ℓ−| LCS(p, q) |).

A TSP path is obtained from the graph, which is "aligned" on the grid, row by row, to form a placement: the TSP is placed from left to right on the first row, right to left on the second, and then alternate in the same way on the remaining rows [29][30][31].



Aligning, row by row of a TSP (solid edges) on a grid.

Algorithm 1 PLACEMENT& EMBEDDING: approximation algorithm for BMP. [31]

Input: ProbesP = {p1, p2, . . . , pn} shall be placed on an array × .

Output: A placement φand an embedding εforP.

Algorithm:

1)Build the fully weighted Gc graph.

2) Find a Q approximate TSP for Gc using the algorithm in [1].

3) Align Q row by row in order to obtain a placement φ.

4) Use the approximation algorithm for P-BMP(that is, by reducing the P-BMP instance to a WMSA instance) to obtain en embedding ε.

Theorem 3: The PLACEMENT & EMBEDDING algorithm is a O(), of approximation for BMP*.*

4.2 Apporximation algorithm BMP –the case of an 1D array

We shall study the special case on an 1D array. Intuitively, the problem is easier than the 2D case. We will show the algorithm that deemonstrates that P-BMP on an 1D array can be optimally solved in polynomial time, while BMP on an 1D array admits an O (1) approximation.

The algorithm to be presented uses a procedure called "Extend". It takes two sequences p and q and a S supersequence of p as input and gives back a supersequence of S and q. Consider c = | LCS (p, q) |, x1, x 2,..., xc indexes of S corresponding to p belonging to LCS (p, q) and y1, y2,..., yc indexes of q belonging to LCS (p, q)

This procedure expands S by inserting characters into q, but not in LCS (p, q): characters between q [yk-1] and q [yk] are inserted just before S [xk] and the characters beyond q [yc ] are added at the end of S. For a clearer exposition of the essence of this procedure we can say that "Extend" keeps track of the indexes of the new S that correspond to q [30] [31] [29].

Theorem 4: The 1 D embedding finds an optimal incorporation for the P-BMP problem on the 1D array in polynomial time [29] [30] [31].

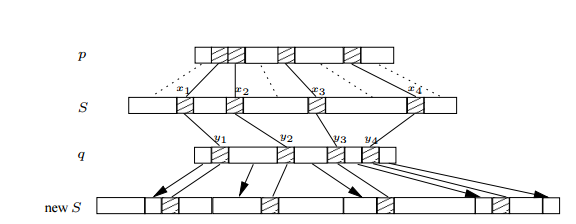


Fig. 4.2:is an illustration of the "Extend" procedure

The shaded squares refer to the LCS characters (p, q). The characters in q, but not in LCS (p, q) are inserted into S so that the order is kept as in q (see arrows) [29] [30] [31.

Algorithm 2 EMBEDDING 1D : approximation algorithm for P- BMP on an 1 D array [31]

Input: Probes P = {p1, p2, . . . , pn}, placed on an 1D array in that order.

Output:Embedding ɛ with a minimum border length.

Algorithm:

1) Set D = p1.

2) For i> 1, use the "Extend" procedure with pi-1, pi and D as input to obtain a new D.

3) For each pi, it is set ɛi so that ɛ [y] = D [y] if D [y] corresponds to the character in pi contained by "Extend", and ɛ [y] = "-" otherwise.

BMP on 1D array. Similar to the case on the 2D array, we will find a placement destination by finding a TSP approximation on the fully weighted Gc graph and then finding a 1D embedding. This algorithm provides a 3/2 approximation for BMP on the 1D array[29][30][31].

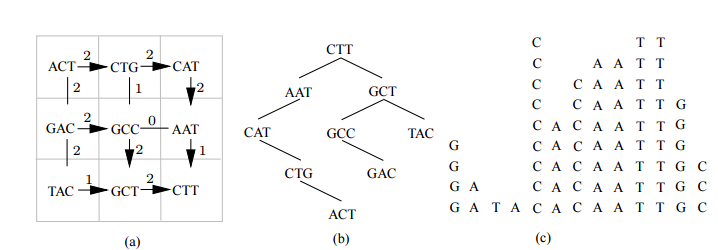
Theorem 5: There is a polynomial time algorithm for BMP on 1D array with an approximation ratio of 3/2 [31].

4.3 Aproximation for P-AMP

In this section you will find the opposite side of the BMP, which we called the agreement maximization problem (AMP). Unlike the BMP, the AMP admits constant approximations, regardless of whether the placement is given in advance or not. BMP can be defined as an agreement maximization problem (AMP) with a different target called "agreement". Minimizing the length of the border is equivalent to maximizing the agreement. And we are able to develop approximation algorithms O (1) for AMP regardless of whether the placement is given in advance or not [31].

We will first present the algorithm studied for the P-AMP, a variant of AMP with a given placement. The AEMBED algorithm.

The AEMBED algorithm uses the "Extend" procedure described above. The order of probes to be considered is determined by a particular T tree with the bottom right probe in G, being the root [31].



The figure represented under letter a) is a set of probes placed on a grid of 3 x 3 G. The values represent the LCS length between two adjacent / neighbor probes. The arrow represented from p to q means parent (p) = q.

In Figure b), it is exemplified the tree built by AEMBED with the root in CTT.

The figure illustrated under letter c) shows how the deposition sequence D iteratively changes. The sequences are put in a way that aligns the characters with the final D sequence.For the construction of the T-tree, a probe parent will be assigned to each probe p, a probe parent, named parent (p). We shall designate r (p) and b (p) as neighbors to the right, respectively neighbors at the bottom of the probe p.

The probes in the right column and the lowest column have r (p) = NULL, respectively b (p) = NULL. We set the parent (p) either r (p) or b (p) depending on whether |LCS (p, r (p))| is higher or if | LCS (p, b (p))| is higher [31] [29] [30].

Below, there is the AEMBED algorithm.

Algorithm for approximation of the P-AMP algorithm (AEMBED) [31]

Input: ProbesP = {p1, p2, . . . , pn} shall be paced on an array × according to placement φ .

Output:An ε embedding for P.

Algorithm:

1) Build a tree T by assigning a parent for each probe p:

if |LCS(p, r(p))| ≥|LCS(p, b(p))| parent(p) = r(p), otherwise parent (p) = b(p).

2) Set D to be the lowest probe on grid G.

3) Cross T for the pre-order: for each crossed p, call the "EXTEND" procedure with parent (p), p and D as input.

4) For each pi, set εi so that ε [y] = D[y] if D[y] corresponds to a character in pi being in EXTEND, otherwise ε [y] = “ − ” .

Thus,AEMBED runs in polynomial time and the difficult part is to find LCS between two sequences.

4.4 Approximation algorithm for AMP

We shall study the general AMP problem to find the placement and the embedding so as to maximize the agreement. It can be shown that the following algorithm has an asymptotic ratio corresponding to 4 [31].

Approximation algorithm for AMP [31]

Input:ProbesP = {p1, p2, . . . , pn}shall be placed on an array × .

Output: A placementφ and an embedding ε for P.

Algorithm:

1)P is partitioned into four disjoint groups, A, C, G, and T: a probe is part of A if the number of A in the probe is maximum over the number of other characters (similarly for C, G and T).

2) Align probes in A group on the array in a row by row manner, followed by aligning the probes in C, G, and T to form the placement φ.

3)For probes in A, align them so that the maximum number of A shall be aligned while the different characters remain not aligned. This forms a partial embedding εa with the Da deposition sequence. Similarly form εc, εc, εt for Dc, Dg, Dt.

4) Combine Da, Dc, Dg şi Dtto obtain D.

5) Extend embedding εa, εc, εg, εt depending on D by entering "−" in the columns corresponding to other groups. Unification of extended embeddings is the resulting embedding ε.

Theorem 6: The asymptotic approximation ratio APLACE & EMBED is 4.

5. Implementation and testing of the algorithm BMP

5.1 Description of BMP algorithm

The BMP problem is solved using the approximation algorithms, as this is an NP-hard problem [29] [30]. The first algorithm for solving BMP is that of Li and his collaborators. It has a lower limit of*O()*, where n is the number of samples that are synthesized. Popa and his collaborators have obtained an algorithm with a reduced upper limit, of*O(n1/4log2n).*Both the first algorithm and the second one are composed of two parts: the placement part and the embedding part. The two algorithms differ only in the placement of samples.

Li's algorithm uses TSP and threading for placement, and for embedding, this step is called P-BMP, this procedure being approximated. The P-BMP refers to the embedding of probes when knowing where they are placed. This problem is reduced to WMSA (weighted minimum sequence alignment) and then to the MRCT (minimum routing cost tree) problem.

Popa's algorithm uses a tree with a metric embedding (tree metric embedding) in order to find sample placement, and this can also be applied to approximate the P-BMP problem. This is possible because the metric used (the LCS distance), which is represented by a linear function, can be introduced into a tree with an upper limit *O (log n)* [29].

Hannenhalli and his collaborators have studied several types of k-threadings to see which ones offer a better result for border length optimization (BL). They have experimentally observed that the BL can be reduced to 20% for a large set of samples if a 1-threading is used compared to the samples arrangement row by row.[32].In the algorithm given by Li, after finding an approximate solution for the TSP problem, the samples are placed in a zigzag manner into the microarray, from right to the left and from left to the right. This is considered to be a 0-threading, where k equals 0 [32]. Popa's algorithm uses a row by row arrangement of the samples.

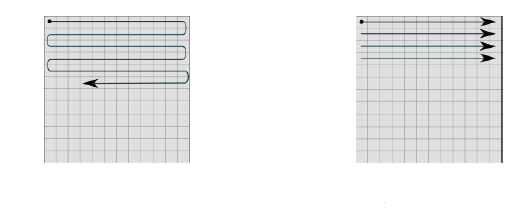


Figure 4.1: On the left it is represented the threding used by Li and his collaborators. On the right there is the threading used by Popa and his collaborators.

5.2 Description of the BMP algorithm implementation

This subchapter presents the algorithms used to solve Li's algorithm and why they were used. Implementation decisions were made on the basis of:

-Functionality of the algorithm: it refers to finding an algorithm that has the best solution so that BL is minimal.

-Complexity of the algorithm: the intention was find a algorithm with as little complexity as possible.

5.2.1 Distance LCS

The LCS or Longest Common Subsequence problem refers to finding a common subsequence which should be maximum for two strings. For example, given the s1 and s2, with s1 = A T C G, and the string s2 = A A C T, then a common maximum subsequence is composed of A C, having a length equal to 2.

The LCS distance denoted by dist (.,.) is a metric used in the presented algorithm to show how nearly two samples, pi and pj, are located one from the other. This is embedded in the program as being the cost of the edge between two samples, where the nodes are associated with the sample set. It is desired that the adjacent samples to the microarray should be as close as possible from the point of view of the LCS distance, since BL is minimized in this manner.

The LCS distance between two samples, pi and pj, is defined as dist (.,.) = li+lj -2|LCS(pi , pj |, where li is the length of sample pi, and lj is the length of sample pj.

Finding the longest common sub-string between two strings can be done by backtracking method, but this process will have an exponential time.

A more efficient solution is achieved with dynamic programming. This will result in a complexity of O (M \* N), where M is the length of the first string, and N the length of the second string. The algorithm is well explained in T.Cormen's book "Introduction to Algorithms".

5.2.2 Travelling Salesman Problem(TSP)

The *Travelling Salesman Problem*is a classic problem in the graph theory [55] [56] [54]. It aims to find the minimum way between cities, so that each one is visited, but only one passage through each. This is transposed into theory by a graph G = (V, E), where V represents the cities, and E the edges, each edge having a cost. In solving the BMP problem, it is assumed that the set of samples is represented by nodes, V respectivey, and the cost attributed to each edge is LCS distance. The TSP problem is a NP-complete problem and to find a solution taking into account each variant it takes a factorial time[55]. This limits the number of samples, since it takes a very long time for calculation. Thus, several algorithms are sought to approximate the solution. An algorithm approximating the TS solution is called "2-approximation" [55] [56] [59]. It has a complexity of O (| E | log | V |). This approximation consists of two steps:

1. Finding a minimum spanning tree (MST)

2. Finding a Euler tour in MST.

Algorithm TSP 2-approximation

1)Input:A graphG(V, E)

2)A MST is sought

3)An Eulerian graph is obtained by doubling all edges

4)An Euler circuit of the graph is obtained by the short-cutting operation of the nodes already visited

5) Output:A cycle that visits each node in V is obtained.

5.2.3Minimum spanning tree(MST)

This problem is formulated through a non-oriented G graph, having V nodes and E edges, each edge having a cost associated. It is desired to find a subgraph that includes all the nodes and a part of the edges, so that the determined subgraph is a tree, and the sum of the costs of the its forming edges is minimal.For the BMP, a complete graph is considered because any sample can be placed next to any other sample if thus reduces the number of masks used to the samples embedding and thus minimize the BL. The cost between the edges is the LCS distance.

In order to find a single MST, the cost between the edges must be different and thus, generally, no unique MSTs for the BMP are obtained. The best known algorithms for finding a MST is that provided by Prim [57], respectively that given by Kruskal [58].We chose the algorithm given by Prim in the BMP implementation because it has a better performance when the number of edges is high, ie | E | ≈ | V | 2. Prim's algorithm functions according to the following principle:

*Given a graph G = (V, E) at each step we choose the cheapest edge leading to an unvisited node and add it to a tree T (Vt, Et), so that no cycles are formed in the tree, and all nodes of G are found in T.*

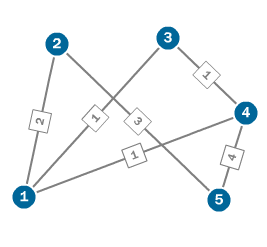
**

Fig. 4.2.2.1:A graph from which a partial minimum cost tree is obtained in the figure 4.2.2.2

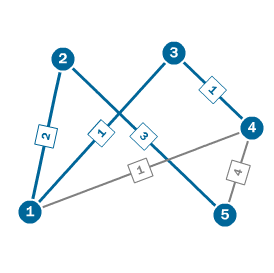


Fig. 4.2.2.2: We obtain a minimum cost tree of the graph (the blue-colored part) represented in fig.4.2.2.1

5.2.4Obtaining an Eulerian circuit

Since we have obtained an MST, in order to have an Eulerian graph it is necessary only to double each of its edges [55].

*Definition (Eulerian Graph):*It is called an Eulerian graph if G is a related graph and each node has an even degree.

*Definition (Tur Euler):*It's called an Eulerian circuit of G, a cycle that visits every node exactly once.

Finding an Euler circuit/cycle in an Eulerian graph by "short-circuiting" is done by browsing the tree skipping the nodes already visited. This is done through an depth-first search.

5.2.5 Minimum Routing Cost Tree (MRCT)

The P-BMP problem is reduced to a WMSA that reduces itself to a MRCT problem [54] [31]. This tree has the following properties:

- for any two nodes, u and v in a tree there is a path between them as well

- the sum of all edges costs on this path is called the "routing cost" of this pair of nodes.

- MRCT refers to minimizing the amount of this cost between any of the pairs of nodes.

Each sequence in WMSA is considered to be a node. The cost of edges is given by the weighted editing distance *w(Si, Sj)\*d(Si, Sj),* where Si and Sj are two sequences of WMSA, and *d(Si, Sj)*  is the LCS distance defined in the previous subchapters. The weight function represented by *w(Si, Sj*) can have only two values:

- 1, where *Siis neighbor toSj*

- 0, otherwise.

5.2.6 Algorithm testing

This subchapter presents a test of Li's and his collaborators' algorithm. The test consists in determining the time in seconds to perform the algorithm requirements. The first test consisted of 25 samples. At the first test, the number of nucleotides for each sample was 3, at the second test was 10, and at the last test the nucleotide number was 20. The data was entered from the keyboard, in a randoom way.

|  |  |
| --- | --- |
| Nucleotides number | Time(s) |
| 3 | 23.493 |
| 10 | 45.636 |
| 20 | 67.167 |

Table 4.2.4.1: For 25 samples

In the second test the number of samples is 50. As in the first table, 3, 10 and 20 nucleotides, respectively, were used per each test.

|  |  |
| --- | --- |
| Nucleotides number | Time(s) |
| 3 | 49.344 |
| 10 | 57.706 |
| 20 | 133.205 |

Tabe 4.2.4.2: For 50 samples

6. Final remarks

In this paper we presented the necessary notions to solve the algorithm of Li and his collaborators in an as short as possible time. Chapter 4 describes in detail the decisions taken to implement it as efficiently as possible.

For the entire algorithm, solution approximation algorithms were used to obtain the placement destination of probes in the microarray and how they are embedded in the chip. For placement, an approximate TSP solution was used, which was reduced to an MST, and then a Euler graph was formed out of which an Euler circuit/cycle was obtained, finally resulting in a cycle that visits each node attributed in this context with the notion of sample.In the case of the P-BMP algorithm, which refers to how the nucleotides are inserted into the microarray after the samples have already a place, there have been approximations as well.

Since the minimal length of the border is requested, i.e. the number of masks used to put the nucleotides in the chip shall be minimal, the algorithm for obtaining P-BMP has been reduced first to the WMSA problem, and then to the MRCT problem [31 ]. Although so many changes have been made to get an optimal solution, it has been noticed that the time taken to solve this problem has not improved significantly.

To solve this problem, other algorithms have also been developed, one of them being that of Popa and his collaborators. They have obtained a complexity*O(n1/4 log2n)*. They have also solved the sample placement in a different way, but the way they embed the probes is the same.

After getting the results, I thought about improving them. As the only changes we have made so far have been in the probe placement, I proposed an idea to change the data embedding, P-BMP, common to both the algorithm proposed by Li and Popa and their collaborators. Since P-BMP can be reduced to a MRCT, then, at a large volume of data,this is approximated by a MST [60][61].

Both placement and sample embedding can be improved. The BMP is a NP-hard problem, thus any satisfactory results obtained here can be successfully used elsewhere. Therefore, this problem deserves further study.

References:

[1] D Gerhold, T Rushmore, and C T Caskey. "Dna chips: promising toys have become powerful tools". Trends Biochem Sci, 24(5):168–73, May 1999.

[2] S P Fodor, J L Read, M C Pirrung, L Stryer, A T Lu, and D Solas. "Light-directed, spatially addressable parallel chemical synthesis". Science, 251(4995):767–73, February 1991.

[3] Ioannis Marcoullis. "Practical verification of an algorithm for Microarray Border Minimisation". September 2011

[4] RL Stears, et al. Nat Med. 2003;9(1):140.Trends in microarray analysis, Nat Med. 2003 Jan; 9(1):140-5.

[5] Tse-Wen Chang, TW (1983). "Binding of cells to matrixes of distinct antibodies coated on solid surface". *Journal of Immunological Methods*. 65 (1–2): 217–23.

 [6]  Schena, M.; Shalon, D.; Davis, R. W.; Brown, P. O. (1995). "Quantitative Monitoring of Gene Expression Patterns with a Complementary DNA Microarray". Science. 270 (5235): 467–70

[7]Wang, D; Carroll, GT; Turro, NJ; Koberstein, JT; Kovác, P; Saksena, R; Adamo, R; Herzenberg, LA; Herzenberg, LA; Steinman, L (2007). "Photogenerated glycan arrays identify immunogenic sugar moieties of Bacillus anthracis exosporium". *Proteomics*. 7 (2): 180–184

[8][Saravanakumar Selvaraj](https://www.ncbi.nlm.nih.gov/pubmed/?term=Selvaraj%20S%5BAuthor%5D&cauthor=true&cauthor_uid=21584183); .[Jeyakumar Natarajan](https://www.ncbi.nlm.nih.gov/pubmed/?term=Natarajan%20J%5BAuthor%5D&cauthor=true&cauthor_uid=21584183). "Microarray Data Analysis and Mining Tools". PMC.

[9][Stephanie A Booth](https://www.ncbi.nlm.nih.gov/pubmed/?term=Booth%20SA%5BAuthor%5D&cauthor=true&cauthor_uid=18159302), DPhil, [Michael A Drebot](https://www.ncbi.nlm.nih.gov/pubmed/?term=Drebot%20MA%5BAuthor%5D&cauthor=true&cauthor_uid=18159302), PhD, [Graham A Tipples](https://www.ncbi.nlm.nih.gov/pubmed/?term=Tipples%20GA%5BAuthor%5D&cauthor=true&cauthor_uid=18159302), PhD, and [Lai King Ng](https://www.ncbi.nlm.nih.gov/pubmed/?term=Ng%20LK%5BAuthor%5D&cauthor=true&cauthor_uid=18159302), PhD."Application of DNA array technology for diagnostic microbiology". PMC

[10]Singh-Gasson S, Green RD, Yue Y, et al. "Maskless fabrication of light-directed oligonucleotide microarrays using a digital micromirror array".Nat Biotechnol 1999;17:974-8.

[11]. de Saizieu A, Certa U, Warrington J, Gray C, Keck W, Mous J. "Bacterial transcript imaging by hybridization of total RNA to oligonucleotide arrays".Nat Biotechnol 1998;16:45-8.

[12]. Fislage R, Berceanu M, Humboldt Y, Wendt M, Oberender H. "Primer design for a prokaryotic differential display RT-PCR".Nucleic Acids Res 1997;25:1830-5.

[13]. de Saizieu A, Gardes C, Flint N, et al. "Microarray-based identification of a novel streptococcus pneumoniae regulon controlled by an autoinduced peptide".J Bacteriol 2000;182:4696-703.

[14]. Talaat AM, Hunter P, Johnston SA. "Genome-directed primers for selective labeling of bacterial transcripts for DNA microarray analysis".Nat Biotechnol 2000;18:679-82.

[15]. Bean P, Wilson J. "HIV genotyping by chip technology".Am Clin Lab 2000;19:16-7.

[16]. Clement JP, Ranst MV. "The role of rodents in emerging and re-emerging human infections".Infect Dis Rev 2000;2:84-7.

[17]. Wilson M, DeRisi J, Kristensen HH, Imboden P, Rane S, Brown PO, Schoolnik GK."Exploring drug-induced alterations in gene expression in Mycobacterium tuberculosis by microarray hybridization".Proc Natl Acad Sci U S A. 1999 Oct 26; 96(22):12833-8.

[18]. Bresnahan WA, Shenk T. "A subset of viral transcripts packaged within human cytomegalovirus particles".Science 2000;288:2373-6.

[19]. Stingley SW, Ramirez JJ, Aguilar SA, et al. "Global analysis of herpes simplex virus type 1 transcription using an oligonucleotide-based DNA microarray".J Virol 2000;74:9916-27.

[20]. Cummings CA, Relman DA." Using DNA Microarrays to study host-microbe interactions".Emerg Infect Dis 2000;6:513-25.

[21]. Manger ID, Relman DA." How the host 'sees' pathogens: global gene expression responses to infection".Curr Opin Immunol 2000;12:215-8.

[22]. Zhu H, Cong JP, Mamtora G, Gingeras T, Shenk T. "Cellular gene expression altered by human cytomegalovirus: global monitoring with oligonucleotide arrays".Proc Natl Acad Sci USA 1998;95:14470-5.

[23]. Alizadeh AA, Eisen MB, Davis RE, et al. "Distinct types of diffuse large B-cell lymphoma identified by gene expression profiling".Nature 2000;403:503-11.

[24]. Golub TR, Slonim DK, Tamayo P, et al. "Molecular classification of cancer: class discovery and class prediction by gene expression monitoring".Science 1999;286:531-7.

[25]. Xiang C, Young H, Alterson H, et al. "Comparison of cellular gene expression in Ebola-Zaire and Ebola-Reston virus-infected primary human monocytes".Nat Genet 1999;23:82

[26]. Brown PO, Botstein D. "Exploring the new world of the genome with DNA microarrays".Nat Genet1999;21(1 Suppl):33-7.

[27]. Bassett DE Jr, Eisen MB, Boguski MS. Gene expression informatics - it's all in your mine.Nat Genet1999;21(1 Suppl):51-5.

[28]. Segio A de Carvalho and Sven Rahmann." Improving the design of genechip arraysby combining placement and embedding". Comput Syst Bioinformatics Conf, 6:417–27,2007.

[29]. [Alexandru Popa](https://arxiv.org/find/cs/1/au:+Popa_A/0/1/0/all/0/1), [Prudence W.H. Wong](https://arxiv.org/find/cs/1/au:+Wong_P/0/1/0/all/0/1), [Fencol C.C. Yung](https://arxiv.org/find/cs/1/au:+Yung_F/0/1/0/all/0/1). "Hardness and Approximation of The Asynchronous Border Minimization Problem", November 2010.

[30]. [Robert Ganian](https://arxiv.org/find/cs/1/au:+Ganian_R/0/1/0/all/0/1), [Martin Kronegger](https://arxiv.org/find/cs/1/au:+Kronegger_M/0/1/0/all/0/1), [Andreas Pfandler](https://arxiv.org/find/cs/1/au:+Pfandler_A/0/1/0/all/0/1), [Alexandru Popa](https://arxiv.org/find/cs/1/au:+Popa_A/0/1/0/all/0/1). "Parameterized Complexity of Asynchronous Border Minimization".

[31]. Cindy Y. Li, Prudence W. H. Wong, Qin Xin, and Fencol C. C. Yung. "Approximating border length for dna microarray synthesis". In Manindra Agrawal, Ding-Zhu Du, Zhenhua Duan, and Angsheng Li, editors, TAMC, volume 4978 of Lecture Notes in Computer Science, pages 410–422. Springer, 2008.

[32]. Sridhar Hannenhalli, Earl Hubell, Robert Lipshutz, and Pavel A Pevzner. Combinatorial algorithms for design of dna arrays. Adv Biochem Eng Biotechnol, 77:1–19, 2002.

[33]. Sergio A. de Carvalho and Sven Rahmann. "Improving the layout of oligonucleotidemicroarrays: Pivot partitioning". In Philipp Bucher and Bernard M. E. Moret, editors, WABI, volume 4175 of Lecture Notes in Computer Science, pages 321–332. Springer, 2006.

[34]. Sergio A. de Carvalho and Sven Rahmann. "Microarray layout as quadratic assignment problem". In Daniel H. Huson, Oliver Kohlbacher, Andrei N.Lupas,Kay Nieselt, and Andreas Zell, editors, German Conference on Bioinformatics, volume 83 of LNI, pages 11–20. GI, 2006.

[35]. Andrew B. Kahng, Ion I. Mandoiu, Pavel A. Pevzner, Sherief Reda, and Alexander Zelikovsky. "Scalable heuristics for design of dna probe arrays". Journal of Computational Biology". 11(2/3):429–447, 2004.

[36]. Andrew B. Kahng, Ion I. Mandoiu, Sherief Reda, Xu Xu, and Alexander Zelikovsky. "Computer-aided optimization of dna array design and manufacturing". IEEE Trans.on CAD of Integrated Circuits and Systems, 25(2):305–320, 2006.

[37]. Dr. Sanjeev Singh1 , Megha Kadam Bedekar. " DNA Microarray and its Applications in Disease Diagnosis ". International Journal of Science and Research (IJSR).

[38] Golub TR, Slonim DK, Tamayo P, et al. Molecular classification of cancer: Class discovery and class prediction by gene expression monitoring. Science. 1999;286:531–537.

[39]van’t Veer LJ, Dai HY, van de Vijver MJ, et al. Gene expression profiling predicts clinical outcome of breast cancer. Nature. 2002;415:530–536.

[40] Singh D, Febbo PG, Ross K, et al. Gene expression correlates of clinical prostate cancer behavior. Cancer Cell. 2002;1:203–209.

[41] Wang T, Hopkins D, Schmidt C, et al. "Identification of genes differentially over-expressed in lung squamous cell carcinoma using combination of cDNA subtraction and microarray analysis". Oncogene. 2000;19:1519–1528.

[42] Alon U, Barkai N, Notterman DA, et al. Broad patterns of gene expression revealed by clustering analysis of tumor and normal colon tissues probed by oligonucleotide arrays. Proceedings of the National Academy of Sciences of the United States of America. 1999;96:6745–6750.

[43]  Ramaswamy S, Tamayo P, Rifkin R, et al. Multiclass cancer diagnosis using tumor gene expression signatures. Proceedings of the National Academy of Sciences of the United States of America. 2001;98:15149–15154

[44]  Brachat A, Pierrat B, Brungger A, Heim J. Comparative microarray analysis of gene expression during apoptosis-induction by growth factor deprivation or protein kinase C inhibition. Oncogene. 2000;19:5073–5082.

[45]  Bonner AE, Lemon WJ, You M. Gene expression signatures identify novel regulatory pathways during murine lung development: implications for lung tumorigenesis. Journal of Medical Genetics. 2003;40:408–417

[46] Rhodes DR, Yu JJ, Shanker K, et al. Large-scale meta-analysis of cancer microarray data identifies common transcriptional profiles of neoplastic transformation and progression. Proceedings of the National Academy of Sciences of the United States of America. 2004;101:9309–9314.

[47]Brachat A, Pierrat B, Xynos A, et al. A microarray-based, integrated approach to identify novel regulators of cancer drug response and apoptosis. Oncogene. 2002;21:8361–8371.

[48] Cutler DJ, Zwick ME, Carrasquillo MM, et al. High-throughput variation detection and genotyping using microarrays. Genome Research. 2001;11:1913–1925

[49] Yan PS, Chen CM, Shi HD, et al. Dissecting complex epigenetic alterations in breast cancer using CpG island microarrays. Cancer Research. 2001;61:8375–8380.

[50] Pollack JR, Perou CM, Alizadeh AA, et al. Genome-wide analysis of DNA copy-number changes using cDNA microarrays. Nature Genetics. 1999;23:41–46.

[51]  Relogio A, Ben-Dov C, Baum M, et al. Alternative splicing microarrays reveal functional expression of neuron-specific regulators in Hodgkin lymphoma cells. Journal of Biological Chemistry. 2005;280:4779–4784.

[52] [Victor Trevino](https://www.ncbi.nlm.nih.gov/pubmed/?term=Trevino%20V%5BAuthor%5D&cauthor=true&cauthor_uid=17660860),  [Francesco Falciani](https://www.ncbi.nlm.nih.gov/pubmed/?term=Falciani%20F%5BAuthor%5D&cauthor=true&cauthor_uid=17660860),   [Hugo A. Barrera-Saldaña](https://www.ncbi.nlm.nih.gov/pubmed/?term=Barrera-Salda%26%23x000f1%3Ba%20HA%5BAuthor%5D&cauthor=true&cauthor_uid=17660860)."DNA microarrays: a powerful genomic tool for biomedical and clinical research".

[53] Dana Lica, Mircea Pașoi. "Fundamentele programării: culegere de probleme de informatică- Pascal și C++ pentru clasa a XI-a".

[54] [Clifford Stein](http://www.okian.ro/autor-615601-clifford-stein.html) , [Thomas H. Cormen](http://www.okian.ro/autor-745910-thomas-h-cormen.html) , [Charles E. Leiserson](http://www.okian.ro/autor-745911-charles-e-leiserson.html) ,[Ronald L. Rivest](http://www.okian.ro/autor-745912-ronald-l-rivest.html). "Introduction to Algorithms"

[55]Christos H. Papadimitriou and Kenneth Steiglitz." Combinatorial Optimization : Algorithms and Complexity" . Dover Publications, July 1998.

[56] Joan M. Aldous and Robin J. Wilson. "Graphs and Applications: An Introductory Approach" .Springer Verlag, 2000.

[57]Prim, R. C. "Shortest connection networks and some generalizations". Bell Systems Technology Journal, 36:1389–1401, 1957.

[58]J. B. Kruskal." On the shortest spanning subtree of a graph and the travelling salesman problem". In Proc. Am. Math. Soc., pages 48–50, February 1956. Published as Proc. Am.Math. Soc., volume 7, number 1.

[59] Vijay V. Vazirani. "Approximation algorithms. Springer". 2001.

[60] P. Mieghem, S. Langen, Influence of the link weight structure on the shortest path, Physical Review E 71 (056113) (2005) 1–13.

[61] P. Mieghem, S.M. Magdalena, Phase transition in the link weight structure of networks, Physical Review E 72 (056138) (2005) 2–7.

[62] Roland B Stoughton."Applications of dna microarrays in biology". Annu Rev Biochem,74:53–82, 2005.

[63] Lance Fortnow, "The Status of the P Versus NP Problem". Communications of the ACM, Vol. 52 No. 9, Pages 78-86, September 2009.

[64] Dorit S. Hochbaum, editor." Approximation Algorithms for NP-Hard Problems". PWS Publishing Company, 1997.

[65] Philip N. Klein and Neal E. Young. "Approximation algorithms for NP-hard optimization problems".CRC Handbook on Algorithms, pages 34-1–34-19. CRC Press, 1999.

[66] Rajeev Motwani, Joseph (Seffi) Naor, and Prabakhar Raghavan. "Randomized approximation algorithms in combinatorial optimization". Dorit Hochbaum,editor, ApproximationAlgorithms for NP-Hard Problems, chapter 11, pages 447–481. PWS Publishing Company, 1997.

[67] Alfred V. Aho, John E. Hopcroft, and Jeffrey D. Ullman. "Data Structures and Algorithms". Addison-Wesley, 1983.