A survey of exact motif finding algorithms

Ali Basim Yousif[1], Hussein Keitan Al-Khafaji[2], Thekra Abbas[1]
[1]Department of Computer Science, College of Science, Mustansiriyah University, Baghdad, Iraq
[2]Department of Computer Communication Engineering, Al-Rafidain University College, Baghdad, Iraq

Article Info

Article history:

Received Dec 20, 2021
Revised Jun 11, 2022
Accepted Jun 23, 2022

Keywords:

Bioinformatics
Data mining
Deoxyribonucleic acid
Motif discovery
Motif mining

ABSTRACT

Deoxyribonucleic acid (DNA) motif finding (discovery/mining) in biological chains is the most recent challenging and interesting trend in bioinformatics. It represents a crucial phase in most bioinformatics systems related to unravelling the secrets of gene functions. Despite the efforts made to date to produce robust algorithms, DNA motif finding remains a difficult task for researchers in this field. In general, biological pattern locating algorithms are categorized into two categories: probabilistic and numerical methods. In this paper, we provide a survey of exact DNA motif finding algorithms and their working principles with a suitable comparison among these algorithms to provide an essential step for researchers in this field.

Corresponding Author:

Ali BasimYousif
Department of Computer Science, College of Science, Mustansiriyah University
Baghdad, Iraq
Email: ali.basim.yousif@uomustansiriyah.edu.iq

1. INTRODUCTION

Bioinformatics is a multidisciplinary field that works at the interaction of biological sciences, statistics and information technology to analysis of genome, sequence information and predict the structure and function of cellular molecules that are used in formation genome. One of the challenging issues in the field of bioinformatics is motif finding/mining/discovery/identification, which is one of the researchers' intriguing sequence analysis concerns [1].

A deoxyribonucleic acid (DNA) motif is a brief, repeating pattern of nucleotides inside a DNA sequence that serves a variety of biological purposes. A brief, repetitive, recurring sequence of nucleotides with biological significance is referred to as a DNA motif. Sequence motifs, often known as regulatory elements, may be found in eukaryotic genes' regulatory regions (RR). These patterns are crucial for identifying transcription factor binding sites (TF-BSs), which aids in understanding how genes are regulated [2]. While intergenic areas are very extensive and very varied, sequence motifs are frequently repeated and preserved, making it difficult to identify them. Therefore, the algorithms in this field vary significantly in the execution times, the amount of memory consumed, and the accuracy of the answers [3].

When there is a big problem (such as DNA motif finding in DNA sequences) and its solution has important applications, that problem will be the focus of researchers for several years to find a viable, if not optimal, solution. The initiation of finding a new solution to the problem of DNA motif discovery in its huge databases requires both knowledge of the nature of the problem and knowledge of previously implemented algorithms. This cannot be accomplished if there are no significant survey studies that collect, make available, and simplify the analysis of previous algorithms. According to our search, which spanned quite a long time, we did not find a survey covering the period of scientific research to find an efficient DNA motif discovery algorithm, which started in earnest in the 1990s. Therefore, this paper aims to achieve several goals as it combines the concepts of bioinformatics, DNA, ribonucleic acid (RNA), and protein structures, methods of representation, and the general architecture of DNA motif discovery systems; and finally, a main goal is to review all the important algorithms in this field to be a comprehensive platform for researchers. These objectives will put their mark on the structure and composition of this paper, as will be noted in the remaining sections of the paper.

2. DNA, RNA AND PROTEIN COMPOSITIONS

DNA is a molecule composed of polynucleotide chains that coil round every different to shape a double helix. DNA keeps the genetic coding for an organism in four arranged bases. DNA chemically consists of a phosphate, sugar and one of four nucleotides of guanine (G), cytosine (C), adenine (A) and thymine (T). There are three bonds of hydrogen between (C, G) pair while twice hydrogen bonds between (A, T) pair. Ribonucleic acid (RNA) is a chain of nitrogen bases, compared to DNA, RNA nitrogen bases include rings of ribose rather than deoxyribose and uracil (U) rather than thymine (T). RNA is transcribed from DNA by RNA polymerase (enzyme) and then processed by other proteins [2].

Proteins are complex and big molecules that perform multiple key functions in the organism. There are 20 distinctive kinds of amino acids that may be combined into proteins which are adenine (A), thymine (T), cytosine (C), guanine (G), isoleucine (I), phenylalanine (F), serine (S), glutamine (Q), histidine (H), asparagine (N), aspartic acid (D), arginine (R), glutamic acid (E), lysine (K), leucine (L), valine (V), tryptophan (W), tyrosine (Y), methionine (M) and proline (P) [3].

Gene expression is the technique via which the genetic coding and the nucleotide series of a gene are used to direct protein synthesis and convey the cell structures. The procedure of the gene expression includes two essential levels which are the transcription and the translation. Transcription is the technique of copying a section of DNA into RNA. The sections of DNA are copied into RNA molecules that can encode proteins are sections to supply messenger RNA (mRNA). Other sections of DNA are copied into RNA molecule known as non-coding RNAs. Translation can be regarded as the entire procedure of ordering and joining the amino acids in a protein depending on the interpretation of an mRNA base sequence. Figure 1 displays the transcription and translation of gene expression [4].

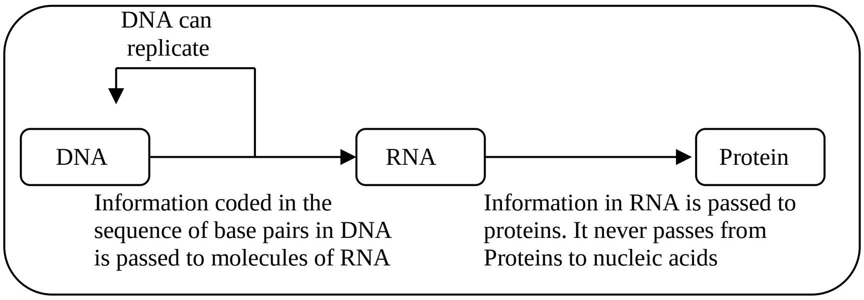

DNA can replicate
DNA
RNA
Protein
Information coded in the sequence of base pairs in DNA is passed to molecules of RNA
Information in RNA is passed to proteins. It never passes from Proteins to nucleic acids

Figure 1. Transcription and translation of gene expression [4]

The motifs are common sequences of DNA, RNA, or protein bases that imply the presence of biological characteristics. Motifs might be constituted in DNA, RNA and protein chains [5]. There are many types of motifs based on the aspects used for categorization. In this paper, we'll concentrate on two kinds of motifs simple/monad and structure which are produced depending on the arrangement of the motif template. The simple motif does not contain gaps and it is easy to discovery in comparison with structure motifs. The structure motif contains gaps and be more complex than simple motif [6].

Motif template is containing a series of nucleotides and gaps which include lower and upper numbers of 'don't care' bases. A simple motif composed of bases according to the sequence under consideration; DNA, RNA or protein sequence. The general structure of motif template is represented by the following backus naur form (BNF) [7]:

$$S1\{[l1,u1]S2\{[l2,u2]S3\}\{.Sn[ln,un]\}Sn+1\} \quad (1)$$

consider the following example for protein motif: KVVVKMKMMMQ [9], [10], AVCCWWE [6], [8] EC. It is a complex motif which includes triple simple motifs of protein bases which match the following pattern, S1{[l1, u1] S2 {[l2, u2]S3 such that:

- S1 is KVVVKMKMMMQ, i.e., a simple motif consisting of 11 bases.
- S2 is AVCCWWE, i.e., a simple motif consisting of 7 bases.
- S3 is EC, i.e., a simple motif consisting of 2 bases.
- [9], [10] is a gap; the distance between two simple motifs, its lower number of unspecified bases is 9bases and its upper number of unspecified bases is 10 bases.
- [6], [8] is the second gap; the distance between two simple motifs, its lower number of unspecified bases is 6 bases and its upper number of unspecified bases is 8 bases.

The lower limit is an integer number that must be less or equal upper bound. Motif templates usually used as input for motif discovery systems to find the motifs matching the entered template. Motif representation can be one of two common methods; the string-based representation and matrix-based representation. The string-based representation is named pattern or consensus. The second one includes position frequency matrix (PFM), position weight matrix (PWM) or profile. In consensus string, the most repeated nucleotide in each location of the consensus sequence is provided. The four DNA bases, (A, C, G, T) are extended to IUPAC characters. For example, the sequence "AATRNG" is a consensus where "R" means a purine (A or G) and "N" means any base. Table 1 and Table 2 depict the IUPAC base codes [8].

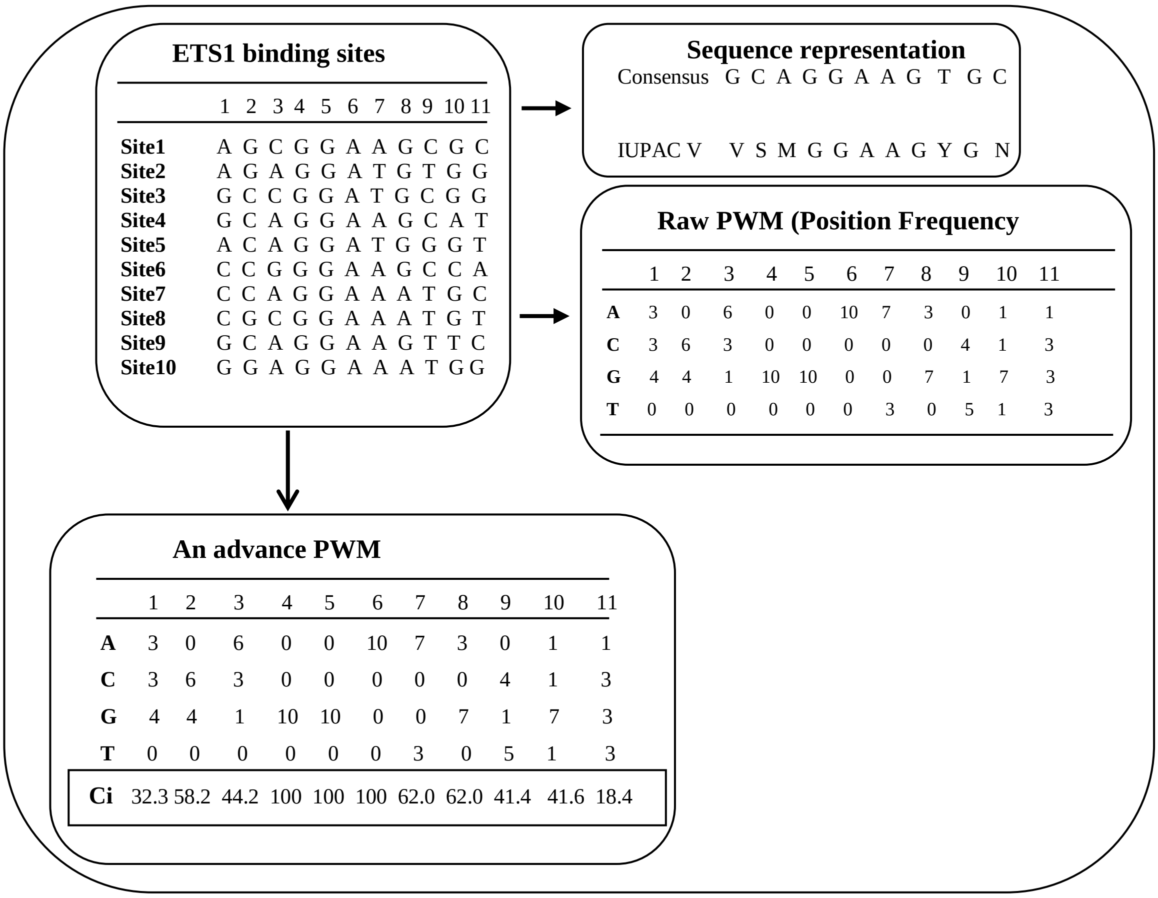
Table 1. The IUPAC nucleotide code with corresponding DNA bases

The IUPAC nucleotide code	The Base
A	Adenine
C	Cytosine
G	Guanine
T (or U)	Thymine (or Uracil)
R	A or G
Y	C or T
S	G or C
W	A or T
K	G or T
M	A or C
B	C or G or T
D	A or G or T
H	A or C or T
V	A or C or G
N	any base

Table 2. Amino acid codes (IUPAC)

The IUPAC Amino Acid code	The Amino Acid
A	Alanine
C	Cysteine
D	Aspartic Acid
E	Glutamic Acid
F	Phenylalanine
G	Glycine
H	Histidine
I	Isoleucine
K	Lysine
L	Leucine
M	Methionine
N	Asparagine
P	Proline
Q	Glutamine
R	Arginine
S	Serine
T	Threonine
V	Valine
W	Tryptophan
Y	Tyrosine

Positional weight matrix (PWM) is a method that used for the illustration of motifs in biological strings. It is a 2D matrix where each location contains a binding site or a motif identifier. The values inside the matrix deliver the chance of every character on the position of each inside a listing of motif positions. Figure 2 presents an example of popular models for motif representing [9].



ETS1 binding sites

	1	2	3	4	5	6	7	8	9	10	11
Site1	A	G	C	G	G	A	A	G	C	G	C
Site2	A	G	A	G	G	A	T	G	T	G	G
Site3	G	C	C	G	G	A	T	G	C	G	G
Site4	G	C	A	G	G	A	A	G	C	A	T
Site5	A	C	A	G	G	A	T	G	G	G	T
Site6	C	C	G	G	G	A	A	G	C	C	A
Site7	C	C	A	G	G	A	A	A	T	G	C
Site8	C	G	C	G	G	A	A	A	T	G	T
Site9	G	C	A	G	G	A	A	G	T	T	C
Site10	G	G	A	G	G	A	A	A	T	G	G

Sequence representation

Consensus G C A G G A A G T G C

IUPAC V V S M G G A A G Y G N

Raw PWM (Position Frequency

	1	2	3	4	5	6	7	8	9	10	11
A	3	0	6	0	0	10	7	3	0	1	1
C	3	6	3	0	0	0	0	0	4	1	3
G	4	4	1	10	10	0	0	7	1	7	3
T	0	0	0	0	0	0	3	0	5	1	3

An advance PWM

	1	2	3	4	5	6	7	8	9	10	11
A	3	0	6	0	0	10	7	3	0	1	1
C	3	6	3	0	0	0	0	0	4	1	3
G	4	4	1	10	10	0	0	7	1	7	3
T	0	0	0	0	0	0	3	0	5	1	3
Ci	32.3	58.2	44.2	100	100	100	62.0	62.0	41.4	41.6	18.4

Figure 2. Motifs representing forms [9]

3. MOTIF DISCOVERY

Motif discovery/motif mining in the sequences of biological data is defined as the process of finding one or more of sequence components, (‘motifs’) in nucleotide sequence which have shared biological operations and activities. It is interesting problem for researchers due to its importance in many bioinformatics applications such as transcription factor binding site (TFBS) [10]-[13]. Motif discovery process is usually divided into three modules/stages; data preprocessing, motif mining, and post-processing.

3.1. Data preprocessing

After the process of obtaining the datasets and motif template, the preprocessing step has two ramifications; motif template preprocessing and dataset preprocessing. Motif template preprocessing is conducted by performing two processes which are parsing of motif template and removing typo errors. Khafaji and Kassim [12], presented techniques to preprocess datasets that dedicated for motif mining such as adjusting the grammar of motif template before mining session in addition to visualize the motifs or datasets. Datasets may require coding process or changing from one representation to another according to the specification of mining algorithm. Some of the bio data representations include extra data such as IDs, and comments, that need some operations of preprocessing before discovery process.

3.2. Motif mining

In this stage, an algorithm for motif discovery should be adopted and according to the adopted algorithm the method of representing the dataset and mined motifs will be determined such as consensus string, and positional weight matrix. The motif mining algorithm will discover the motifs hidden in the database according to the constraints of the motif template. This paper concentrates on the motif mining algorithms.

3.3. Post-processing

Post-processing, also called post-mining, reporting, and visualization, is the last stage that evaluates and presents the results of motif discovery in forms that can be utilized by the decision maker, such as tables, graphs, charts, and/or colored text. Post-processing can combine one or more visualization methods. Most of the results of the motif discovery system are presented as a record for each discovered motif. This record consists of many fields, such as the basis of the motif sequence, motif length, and the start and last position of the motif in the dataset. Figure 3 illustrates a general architecture for motif discovery systems, including the three stages. According to the mentioned figure, the pre-processing of datasets can be executed separately to manipulate the available datasets that will be used in the future mining session. Motif template preprocessing is an interactive process related to the entered user template.

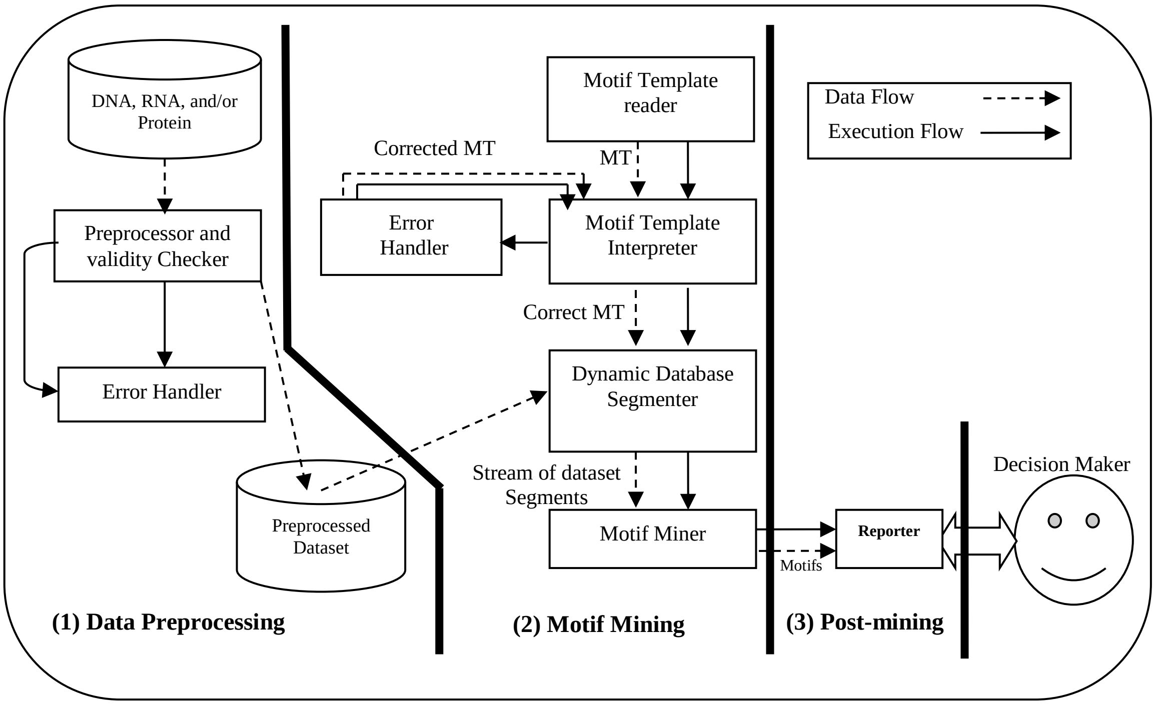


Figure 3. General architecture of the motif discovery system [10]

4. EXACT DNA MOTIF DISCOVERY ALGORITHMS

Mainly, there are two kinds of algorithms for motif discovery; the probabilistic approach and the enumerative approach. The motif mining algorithms related to the first approach usually need a small number of arguments. Furthermore, they depend on base distributions for the sites that are available in binding space to determine the existence of the motif. In the enumeration approach, the discovery process depends on calculating word similarities in consensus sequences [14]. Exact motif mining algorithms usually belong to the enumeration approach. Table 3 presents a number of algorithms that are concerned with this topic and for a period of time that is not short. The Researches conducted during the period 1998 to 2005 are almost similar in behavior and data structures used. Table 3 lists these algorithms with the reference numbers [15]-[24].

The Voting algorithm Chin and Leung [25] is one of the distinguished algorithms such that it manipulates unprecedented motif lengths that the previous algorithms did not manipulate due to the time and space consumed for the mining process. The algorithm works admirably for synthesis and the actual dataset. Sze and Zhao [26] produced an improved pattern-driven algorithm that assures finding the significant motifs in a time that depends on the length of the motif to be mined and the size of the DNA sequence under mining.

A qualitative leap in motif mining algorithms occurred with the design of the algorithm called SMOTIF that regarded as one of the early works in this field. It is designed by Zhang and Zaki [27], SMOTIF is robust algorithm to discover structured motifs for one or more sequences. The algorithm can efficiently search for both patterns and profiles. SMOTIF can search for wide interval gaps, long terminal repeats (LTR) retrotransposons and find potential composite transcription factor binding sites.

Zhang and Zaki [28], proposed an efficient algorithm to mine structured motifs called EXMOTIF according to a given sequence in addition to the motif template. It discovers all occurrences of structure motifs that have quorum q. The experiments showed that EXMOTIF is efficient in terms of both time and space and outperforms RISO algorithm which proposed by Carvalho *et al.* [30], in mining the single or complex transcription factor binding sites (TFs). EXMOTIF is a robust algorithm for discovering complex motifs, especially in DNA chains.

Halachev and Shiri [33], studied the features of genomes and proposed an efficient algorithm to discover structural motifs named exact match, overlapping structured motif search (EMOS). It depends on the suffix tree index. Numerous experiments were performed to evaluate EMOS. To evaluate EMOS and compare its performance with the SMOTIF algorithm [27], many cases were attempted, in some of which its performance was comparable to SMOTIF, but in most cases the search time of EMOS was faster than SMOTIF.

Sharov and Ko [34], designed an algorithm named CisFinder. It deals with large chains of more than 50 Mb with considerable processing speed, especially when the desired motif is short. To operate CisFinder, the nucleotide replacement matrix for each n-mer word should be determined in addition to the construction of PFMs. To generate non-redundant motifs, the PFM extends across adjacent base and gap areas, followed by a clustering process. However, the characteristics of CisFinder can be summarized as shown in: i) It extracts all represented motifs and describes them with PFMs; ii) It can successfully manipulate long chains; iii) Because of its faster motif discovery execution time, it works interactively and runs the analysis several times after resetting the parameters; iv) It plays its role by reducing the enrichment of DNA motifs. These characteristics are the main factors to making CisFinder superior to the PMS3P algorithm which proposed by Sharma and Rajasekaran [35], and RecMotif algorithm which proposed by Sun *et al.* [36].

Sun *et al.* [37], proposed the so-called ListMotif, a sample-driven algorithm that creates a list of motif instances using substrings from the data. ListMotif is memory efficient and time efficient by avoiding recalculation of the Hamming distance between substrings. The results of synthetic data tests show that ListMotif can detect long and weak motifs compared to some previously proposed algorithms.

Kuksa and Pavlovic [38], proposed stemming algorithm to find motifs in sequences. The algorithm reduces computational complexity shows a powerful run-time improvement compared to current motif search algorithms MITRA which proposed by Eskin and Pevzner [19], and PMS Prune which proposed by Pisanti *et al.* [29], and RISOTTO which proposed by Davila *et al.* [31], for long motifs. The proposed algorithm can be applied to other cases and difficult problems in DNA sequences analysis.

Bailey [39], produced an algorithm named discriminative regular expression motif elicitation (DREME). DREME can locate short and middle DNA motifs of eukaryotic transcription factors. Also, it is optimized to search extremely big ChIP-seq databases in minutes therefore it is regarded as quick and scalable algorithm. It includes two loops; to discover many non-redundant motifs in a set of chains, the external loop specifies the most important motifs through heuristic motif searches, and the best motifs found replace their appearance with special characters. The search process is then repeated many times until the value of the new motif falls below the determined significance threshold.

Sun *et al.* [40], proposed a tree-based motif discovery algorithm (tree motif) capable of detecting longer motifs than existing methods in terms of accuracy and execution time. A Tree Motif transforms the graphic representation of a motif into a tree-structured representation. In this representation, the tree that branches at each node in each sequence represents the motif instance. The tree construction method is based on the discovery of novel motifs. Tree motif performance has been demonstrated in both synthetic and real biological data. Provable algorithm which proposed by Chen and Wang [41], depends on generating a wide range of candidates, and then it makes a pruning process to exclude improvable candidates to be discovered motif according to the restrictions of the motif template. PMS4 algorithm which proposed by Rajasekaran and Dinh [42], and PMS5 algorithm which proposed by Dinh *et al.* [43], and PMS6 algorithm which proposed by Bandyopadhyay *et al.* [44], were designed depending on the search tree data structure. Therefore, their performance is described by the term of the difference in traversing the tree and its size in memory.

Most of the mentioned algorithms in this paper calibrate themselves for parallel processing with the need to make modifications to their steps and possibly the data structures used, but algorithm PairMotif which proposed by Yu *et al.* [45], was originally designed to be implemented in an environment that supports parallelism in terms of hardware of the computer system and software components which support multi-thread sytem. The qPMS7 algorithm which proposed by Dinh *et al.* [46], was developed depending on search tree data structure. Nicolae and Rajasekaran [47], designed the PMS8 algorithm based on the qPMS7 algorithm [46], the PMS8 is a robust algorithm to manipulate the planted motif search (PMS) problem. Its efficiency is obtained from the subtle coding, which involves several speedup services and distinguished memory management depending on cache locality. Another reason for the efficiency of PMS8 algorithm

[47], is its ability to produce neighborhoods for n of l-mers at a time, depending on suggested pruning conditions. PMS8 was compared with qPMS7 algorithm using datasets discussed in [46]. The results showed the equality or the domination of PMS8 algorithm in most experiments.

Table 3. Exact motif finding algorithms

Algorithm	Year	Operating Principle	Reference	Algorithm	Year	Operating Principle	Reference
SPELLER	1998	Suffix tree	[15]	Pampa	2007	tree search	[32]
Spelling	1998	Suffix tree	[15]	EMOS	2008	Suffix tree	[33]
TravStrD	2000	tree based	[16]	CisFinder	2009	position frequency matrices (PFMs)	[34]
TravStrR	2000	tree search	[16]	PMS3P	2009	tree search	[35]
WINNOWER	2000	Graph theoretic	[17]	RecMotif	2010	Reference sequence/vertex	[36]
SMILE	2002	Suffix tree	[18]	ListMotif	2010	Graph theoretic	[37]
MITRA	2003	Prefix tree/mismatch tree and graph	[19]	Stemming	2010	neighborhood generation	[38]
CENSUS	2004	tree search	[20]	DREME	2011	Simple word based	[39]
Weeder	2004	Suffix tree	[21]	TreeMotif	2011	Graph theoretic	[40]
cWINNOWER	2004	Graph theoretic	[22]	Provable	2011	Modified candidate	[41]
PSMILE	2004	Suffix tree	[23]	PMS4	2011	tree search	[42]
PMS1	2005	Radix sorting	[24]	PMS5	2011	tree search	[43]
PMS2	2005	Radix sorting	[24]	PMS6	2012	tree search	[44]
PMS3	2005	Radix sorting	[24]	PairMotif	2012	Parallel computing	[45]
Voting	2005	Clustering	[25]	qPMS7	2012	tree search	[46]
Improved Pattern-driven	2006	pattern-driven approach	[26]	PMS8	2014	Random sampling	[47]
SMOTIF	2006	Inverted index of symbol positions	[27]	FMotif	2014	Suffix tree	[48]
EXMOTIF	2006	Inverted index of symbols and hash table	[28]	SLI-REST	2014	Suffix tree	[49]
RISOTTO	2006	Box links and suffix tree	[29]	qPMSPruneI	2014	tree search	[50]
RISO	2007	suffix tree	[30]	qPMS9	2015	Random sampling	[51]
PMSi	2007	tree search	[31]	qPMS10	2016	Random sampling	[52]
PMSP	2007	tree search	[31]	ET-Motif	2016	Suffix tree	[53]
PMSPrune	2007	tree search	[31]				

Jia *et al*. [48] proposed an algorithm named FMotif, which belongs to the enumeration category of algorithms for extracting motifs from sequences. The role of FMotif was tested using mouse ChIP-seq data sets for 12 deoxyribonucleic acid (DNA) binding TF involved in the pluripotency and self-renewal of mouse embryonic stem cells. Experiments have shown that FMotif has diminished execution time and it is of exact type when searching for motifs in (l, d) chain samples. Also, FMotif has satisfied performance in determining motifs in ChIP-rich areas. Generally, it showed a compromise among time, space, and accuracy. FMotif outperforms many algorithms such as SPELLER algorithm which proposed by Sagot [15], MITRA algorithm which proposed by Eskin and Pevzner [19], and WEEDER algorithm which proposed by Pavesi [21]. However, in spite of its accuracy and speed, it is slower than CisFinder algorithm [34]. FMotif algorithm [48], exchanges predominance in several experiments with SLI-REST algorithm which proposed by Cazaux and Rivals [49].

Nicolae and S. Rajasekaran [51], preduced an efficient random algorithm for solving planted motif search (PMS) problemswhich is called qPMS9 [51]. The modification of qPMS9 leads to qPMS10 algorithm which proposed by Xiao *et al*. [52]. The qPMS10 deals with challenging (l,d)-motifs. It is a non-deterministic algorithm, therefore theoretical analysis shows that qPMS10 algorithm [52], is very reasonable compared to qPMS9 algorithm [51]. The experimental results depict its scalability to manipulate growing databases and its efficiency to mine large datasets.

Al-Okaily and Huang [53], proposed an algorithm called ET-motif which depends mainly on a novel data structure named Error tree, which used for hamming distance and wildcards matching in DNA sequences. The proposed tree excluded the time and space required to balance the suffix tree used in some algorithms. ET-motif concentrates on reducing the space and time of motif mining process that are actually reduced according to specified factors.

5. DISCUSSION

As sequencing technology advances, the amount of biological sequence data in public databases has increased, making the discovery of motifs increasingly important in computer science and molecular biology.

Finding a motif presents some difficulties such as the motifs are not the same, the position of the motif is unknown, the arrangement of motif is unknown and the position of one motif in each arrangement is irrelevant to other motifs. There are two types of motif discovery algorithms that are enumerative approach and probabilistic approach. An enumerated approach depends on the description of a particular motif model counts and compares the frequencies of oligonucleotides in all possible motifs. This has several advantages such as the global optimality, short motifs, it helps to find motifs in the eukaryotic genome and the data structure is optimized, so the speed is high. The problems with this approach are common for example the transcription aspect motifs frequently have a few weakly restrained positions that want to be post-processed with the aid of using a few clustering systems, long time processing is another conflict because it tests each feasible substring within the enter dataset, there are multiple error motifs.

There are many algorithms designed based on this approach. YMF is designed for the yeast genome and either cannot recognize long motifs or the number of degenerate positions is large. DREME is an identification motif recognition tool for detecting multiple short, non-redundant, statistically significant motifs in a short amount of time using a simplified form of regular expression words. DREME is compared to the MEME algorithm, and the results show that the DREME algorithm can correctly predict experimental ChIPseq sequence motifs with shorter execution times than MEME. The CisFinder algorithm tested on the TFs ChIPseq data was expressed in ES cells. CisFinder can accurately identify the PFM of the TFs binding motif and is faster than MEME, Weeder and RSAT. CisFinder can find low enhancement motifs, but does not support the output of motifs of a particular length. Weeder algorithm uses suffix trees to speed up word enumeration techniques, but is less efficient for long motifs. The FMotif algorithm can identify the length of unknown motifs in ChIP-rich areas. The graph-based method is the same simple-based method, but represents motif instances through graphs to facilitate search strategies.

6. CONCLUSION

The discovery of motifs is biologically considered important and it is the process of identifying and extracting the patterns needed to understand the complex biological mechanisms of an organism. The various techniques used in the various motif inference tool design paradigms show the growing efforts of researchers to develop efficient algorithms for predicting genomic function. The efficiency of these algorithms is measured by time complexity, which is influenced by the choice of data structure used in the design paradigm. There are two types of motif discovery algorithms which are the probabilistic approach and the enumeration approach. The algorithms are varying by many factors such as speed, memory consumption, the type of data structure used, the guarantee of finding all occurrences of the motif, the type and size of the motif to be discovered, and the size of the database. The enumeration methods are an exhaustive search and it is the only method that ensures finding all motifs. However, they are very slow and require a lot of parameters, therefore they become difficult to deal with either long motifs or big data.

REFERENCES

[1] A. Shanker, “Intellectual property rights and bioinformatics: an introduction,” in *Bioinformatics: Sequences, Structures, Phylogeny*, Singapore: Springer Singapore, 2018, pp. 1–14.

[2] P. Singh and N. Singh, “Role of data mining techniques in bioinformatics,” *International Journal of Applied Research in Bioinformatics*, vol. 11, no. 1, pp. 51–60, Jan. 2021, doi: 10.4018/IJARB.2021010106.

[3] H. K. Al-Khafaji and Z. M. Jameel, “A New Approach to DNA, RNA, and Protein motifs templates visualization and analysis via compilation technique,” *IOSR Journal of Computer Engineering*, vol. 19, no. 02, pp. 15–25, Feb. 2017, doi: 10.9790/0661-1902011525.

[4] R. Jiang, X. Zhang, and M. Q. Zhang, *Basics of Bioinformatics*. USA, Springer, 2013.

[5] M. Kumar et al., “The eukaryotic linear motif resource: 2022 release,” *Nucleic Acids Research*, vol. 50, no. D1, pp. D497–D508, Jan. 2022, doi: 10.1093/nar/gkab975.

[6] G. K. Sandve and F. Drabløs, “A survey of motif discovery methods in an integrated framework,” in *Biology Direct*, vol. 1, 2006, pp. 1–6.

[7] J. Levine, *Introducing Flex and Bison*. Flex & Bison, 2009.

[8] F. A. Hashim, M. S. Mabrouk, and W. Al-Atabany, “Review of different sequence motif finding algorithms,” *Avicenna journal of medical biotechnology*, vol. 11, no. 2, pp. 130–148, 2009.

[9] M. Vahed, M. Vahed, and L. X. Garmire, “BML: a versatile web server for bipartite motif discovery,” *Briefings in Bioinformatics*, vol. 23, no. 1, Jan. 2022, doi: 10.1093/bib/bbab536.

[10] N. Qader and H. K. Al-Khafaji, “Motif discovery and data mining in bioinformatics,” *International Journal of Computers & Technology*, vol. 13, no. 1, pp. 4082–4095, 2014, doi: 10.24297/ijct.v13i1.2932.

[11] F. Bin Ashraf and M. S. R. Shafi, “MFEA: An evolutionary approach for motif finding in DNA sequences,” *Informatics in Medicine Unlocked*, vol. 21, p. 100466, 2020, doi: 10.1016/j.imu.2020.100466.

[12] H. Khafaji and G. Kassim, “A new approach to motif templates analysisvia compilation technique,” *Journal of Al-Rafidain University College for Sciences*, vol. 2, pp. 180–208, 2014, doi: 10.13140/RG.2.2.17048.03841.

[13] Y. He, Z. Shen, Q. Zhang, S. Wang, and D. S. Huang, “A survey on deep learning in DNA/RNA motif mining,” *Briefings in Bioinformatics*, vol. 22, no. 4, Jul. 2021, doi: 10.1093/bib/bbaa229.

[14] M. K. Das and H. K. Dai, “A survey of DNA motif finding algorithms,” *BMC Bioinformatics*, vol. 8, no. SUPPL. 7, p. S21, Dec. 2007, doi: 10.1186/1471-2105-8-S7-S21.

[15] M. F. Sagot, “Spelling approximate repeated or common motifs using a suffix tree,” *in Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 1380, 1998, pp. 374–390, doi: 10.1007/bfb0054337.

[16] H. Al-Shaikhli, “Approximate algorithms for regulatory motif discovery in DNA,” *Dissertations*, pp. 1–115, 2019, [Online]. Available: https://scholarworks.wmich.edu/dissertations/3454.

[17] P. A. Pevzner and S. H. Sze, “Combinatorial approaches to finding subtle signals in DNA sequences,” in *Proceedings International Conference on Intelligent Systems for Molecular Biology ; ISMB. International Conference on Intelligent Systems for Molecular Biology*, 2000, vol. 8, pp. 269–278.

[18] L. Marsan and M. F. Sagot, “Algorithms for extracting structured motifs using a suffix tree with an application to promoter and regulatory site consensus identification,” *Journal of Computational Biology*, vol. 7, no. 3–4, pp. 345–362, Aug. 2000, doi: 10.1089/106652700750050826.

[19] E. Eskin and P. A. Pevzner, “Finding composite regulatory patterns in DNA sequences,” *Bioinformatics (Oxford, England)*, vol. 18, no. Suppl 1, pp. S354–S363, Jul. 2002, doi: 10.1093/bioinformatics/18.suppl_1.S354.

[20] P. A. Evans and A. D. Smith, “Toward optimal motif enumeration,” in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 2748, 2003, pp. 47–58, doi: 10.1007/978-3-540-45078-8_5.

[21] G. Pavesi, P. Mereghetti, G. Mauri, and G. Pesole, “Weeder web: Discovery of transcription factor binding sites in a set of sequences from co-regulated genes,” *Nucleic Acids Research*, vol. 32, no. WEB SERVER ISS., pp. W199–W203, Jul. 2004, doi: 10.1093/nar/gkh465.

[22] S. Liang, “CWINNOWER algorithm for finding fuzzy DNA motifs,” in *Proceedings of the 2003 IEEE Bioinformatics Conference, CSB 2003*, 2003, pp. 260–265, doi: 10.1109/CSB.2003.1227326.

[23] A. M. Carvalho, A. L. Oliveira, A. T. Freitas, and M. F. Sagot, “A parallel algorithm for the extraction of structured motifs,” in *Proceedings of the ACM Symposium on Applied Computing*, 2004, vol. 1, pp. 147–153, doi: 10.1145/967900.967932.

[24] S. Rajasekaran, S. Balla, and C. H. Huang, “Exact algorithms for planted motif challenge problems,” in *Series on Advances in Bioinformatics and Computational Biology*, Jan. 2005, vol. 1, pp. 249–259, doi: 10.1142/9781860947322_0025.

[25] F. Y. L. Chin and H. C. M. Leung, “Voting algorithms for discovering long motifs,” in *Series on Advances in Bioinformatics and Computational Biology*, Jan. 2005, vol. 1, pp. 261–271, doi: 10.1142/9781860947322_0026.

[26] S. H. Sze and X. Zhao, “Improved pattern-driven algorithms for motif finding in DNA sequences,” in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 4023 LNBI, Berlin, Heidelberg: Springer Berlin Heidelberg, 2007, pp. 198–211, doi: 10.1007/978-3-540-48540-7_17.

[27] Y. Zhang and M. J. Zaki, “SMOTIF: Efficient structured pattern and profile motif search,” *Algorithms for Molecular Biology*, vol. 1, no. 1, p. 22, Dec. 2006, doi: 10.1186/1748-7188-1-22.

[28] Y. Zhang and M. J. Zaki, “EXMOTIF: Efficient structured motif extraction,” *Algorithms for Molecular Biology*, vol. 1, no. 1, p. 21, Dec. 2006, doi: 10.1186/1748-7188-1-21.

[29] N. Pisanti, A. M. Carvalho, L. Marsan, and M. F. Sagot, “RISOTTO: Fast extraction of motifs with mismatches,” in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 3887 LNCS, 2006, pp. 757–768, doi: 10.1007/11682462_69.

[30] A. M. Carvalho, A. T. Freitas, A. L. Oliveira, and M. F. Sagot, “An efficient algorithm for the identification of structured motifs in DNA promoter sequences,” *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, vol. 3, no. 2, pp. 126–139, Apr. 2006, doi: 10.1109/TCBB.2006.16.

[31] J. Davila, S. Balla, and S. Rajasekaran, “Fast and practical algorithms for planted (l, d) motif search,” *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, vol. 4, no. 4, pp. 544–552, Oct. 2007, doi: 10.1109/TCBB.2007.70241.

[32] J. Davila, S. Balla, and S. Rajasekaran, “Pampa: An improved branch and bound algorithm for planted (l, d) motif search,” *Springer*, 2007.

[33] M. Halachev and N. Shiri, “Fast structured motif search in DNA sequences,” in *Communications in Computer and Information Science*, vol. 13, Berlin, Heidelberg: Springer Berlin Heidelberg, 2008, pp. 58–73.

[34] A. A. Sharov and M. S. H. Ko, “Exhaustive search for over-represented DNA sequence motifs with cisfinder,” *DNA Research*, vol. 16, no. 5, pp. 261–273, Oct. 2009, doi: 10.1093/dnares/dsp014.

[35] D. Sharma and S. Rajasekaran, “A simple algorithm for (l, d) motif search,” in *2009 IEEE Symposium on Computational Intelligence in Bioinformatics and Computational Biology, CIBCB 2009 - Proceedings*, Mar. 2009, pp. 148–154, doi: 10.1109/CIBCB.2009.4925721.

[36] H. Q. Sun, M. Y. H. Low, W. J. Hsu, and J. C. Rajapakse, “RecMotif: A novel fast algorithm for weak motif discovery,” *BMC Bioinformatics*, vol. 11, no. SUPPL. 11, p. S8, Dec. 2010, doi: 10.1186/1471-2105-11-S11-S8.

[37] H. Q. Sun, M. Y. H. Low, W. J. Hsu, and J. C. Rajapakse, “ListMotif: A time and memory efficient algorithm for weak motif discovery,” in *Proceedings of 2010 IEEE International Conference on Intelligent Systems and Knowledge Engineering, ISKE 2010*, Nov. 2010, pp. 254–260, doi: 10.1109/ISKE.2010.5680875.

[38] P. P. Kuksa and V. Pavlovic, “Efficient motif finding algorithms for large-alphabet inputs,” *BMC Bioinformatics*, vol. 11, no. SUPPL. 8, p. S1, Oct. 2010, doi: 10.1186/1471-2105-11-S8-S1.

[39] T. L. Bailey, “DREME: Motif discovery in transcription factor ChIP-seq data,” *Bioinformatics*, vol. 27, no. 12, pp. 1653–1659, Jun. 2011, doi: 10.1093/bioinformatics/btr261.

[40] H. Q. Sun, M. Y. H. Low, W. J. Hsu, C. W. Tan, and J. C. Rajapakse, “Tree-structured algorithm for long weak motif discovery,” *Bioinformatics*, vol. 27, no. 19, pp. 2641–2647, Oct. 2011, doi: 10.1093/bioinformatics/btr459.

[41] Zhi-Zhong Chen and Lusheng Wang, “Fast exact algorithms for the closest string and substring problems with application to the planted (L,d)-motif model,” *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, vol. 8, no. 5, pp. 1400–1410, Sep. 2011, doi: 10.1109/TCBB.2011.21.

[42] S. Rajasekaran and H. Dinh, “A speedup technique for (l, d)-motif finding algorithms,” *BMC Research Notes*, vol. 4, no. 1, p. 54, Dec. 2011, doi: 10.1186/1756-0500-4-54.

[43] H. Dinh, S. Rajasekaran, and V. K. Kundeti, “PMS5: An efficient exact algorithm for the (ℓ, d)-motif finding problem,” *BMC Bioinformatics*, vol. 12, no. 1, p. 410, Dec. 2011, doi: 10.1186/1471-2105-12-410.

[44] S. Bandyopadhyay, S. Sahni, and S. Rajasekaran, “PMS6: A fast algorithm for motif discovery,” *International Journal of Bioinformatics Research and Applications*, vol. 10, no. 4–5, pp. 369–383, 2014, doi: 10.1504/IJBRA.2014.062990.

[45] Q. Yu, H. Huo, Y. Zhang, and H. Guo, “PairMotif: A new pattern-driven algorithm for planted (l, d) dna motif search,” *PLoS ONE*, vol. 7, no. 10, p. e48442, Oct. 2012, doi: 10.1371/journal.pone.0048442.

[46] H. Dinh, S. Rajasekaran, and J. Davila, “qPMS7: A fast algorithm for finding (ℓ, d)-motifs in DNA and protein sequences,” *PLoS ONE*, vol. 7, no. 7, p. e41425, Jul. 2012, doi: 10.1371/journal.pone.0041425.

[47] M. Nicolae and S. Rajasekaran, “Efficient sequential and parallel algorithms for planted motif search,” *BMC Bioinformatics*, vol. 15, no. 1, p. 34, Dec. 2014, doi: 10.1186/1471-2105-15-34.

[48] C. Jia, M. B. Carson, Y. Wang, Y. Lin, and H. Lu, “A new exhaustive method and strategy for finding motifs in ChIP-enriched regions,” *PLoS ONE*, vol. 9, no. 1, p. e86044, Jan. 2014, doi: 10.1371/journal.pone.0086044.

[49] B. Cazaux and E. Rivals, “Reverse engineering of compact suffix trees and links: A novel algorithm,” *Journal of Discrete Algorithms*, vol. 28, pp. 9–22, Sep. 2014, doi: 10.1016/j.jda.2014.07.002.

[50] S. Tanaka, “Improved exact enumerative algorithms for the planted (l, d)-motif search problem,” *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, vol. 11, no. 2, pp. 361–374, Mar. 2014, doi: 10.1109/TCBB.2014.2306842.

[51] M. Nicolae and S. Rajasekaran, “QPMS9: An efficient algorithm for quorum planted motif search,” *Scientific Reports*, vol. 5, no. 1, p. 7813, Jul. 2015, doi: 10.1038/srep07813.

[52] P. Xiao, S. Pal, and S. Rajasekaran, “QPMS10: A randomized algorithm for efficiently solving quorum Planted Motif Search problem,” in *Proceedings - 2016 IEEE International Conference on Bioinformatics and Biomedicine, BIBM 2016*, Dec. 2017, pp. 670–675, doi: 10.1109/BIBM.2016.7822598.

[53] A. Al-Okaily and C. H. Huang, “ET-motif: solving the exact (l, d)-planted motif problem using error tree structure,” *Journal of Computational Biology*, vol. 23, no. 7, pp. 615–623, Jul. 2016, doi: 10.1089/cmb.2015.0238.

BIOGRAPHIES OF AUTHORS

Ali Basim Yousif received the B.Sc. degree in computer science from Thi-Qar University, Thi-Qar, Iraq, in 2007 and M.Sc. degreefrom Basrah University, Basrah, Iraq, in 2013. He became an Lecturer in 2016. Currently, he is a Ph.D. student in Department of Computer Science, College of Science, Mustansiriyah University, Baghdad, Iraq. He has published 2 papers in refereed journal. His research interests include the bioinformatics, data mining, data science and artificial intelligence. He can be contacted at email: ali.basim.yousif@uomustansiriyah.edu.iq.

Hussein Keitan Al-Khafaji received the B.Sc., M.Sc. and Ph.D. degrees from University of Technology, Baghdad, Iraq, in 1989, 1992, 2002 respectively. He has been a professor of AI and Data Mining in Al-Rafidain University College, since 2012. He is currently the Head of computer communications Eng. Dept., Al-Rafidain University College, Baghdad, Iraq. Also he is a member of the council of Iraqi Commission for Computers and Informatics in the Ministry of Higher Education and Scientific Research in Iraq. He has published more than 70 refereed journal and conference papers in the fields of AI and Data Mining and its applications. He can be contacted at email: hussain.ketan.elc@ruc.edu.iq.

Thekra Abbas received the B.Sc. degree in computer science from the University of technology, Iraq, the M.Sc. degree in computer science from Mustansiriyeah University, college of science computer science department, Iraq, and the Ph.D. degree in computer science from Centeral South University, China. She used to hold several administrative posts with the Computer Science Department, Mustansirieyah University, from 2000 to 2021, including the Head of Department of Computer Science. She has supervised and co-supervised more than 10 masters and 3 Ph.D. students. She has authored or coauthored more than 20 publications: 4 proceedings and 16 journals, with 4 H-index and more than 30 citations. Her research interests include Multimedia, Data mining, machine learning, and intelligent systems. She can be contacted at email: thekra.abbas@uomustansiriyah.edu.iq.