**CHAPTER -1**

**INTRODUCTION**

* 1. **Introduction**

“An idea that is not dangerous is unworthy of being called an idea at all.”

-Oscar Wilde

Bioinformatics is a knowledge domain field that develops strategies and package tools for understanding biological information. As A knowledge domain field of science, bioinformatics combines technology, statistics, arithmetic, and engineering to investigate and interpret biological information. Bioinformatics has been used for in semiconductor analyses of biological queries victimization mathematical and applied math techniques.

Bioinformatics is each an umbrella term for the body of biological studies that use computer programming as a part of their methodology, similarly as a relevancy specific analysis "pipelines" that area unit repeatedly used, notably within the field of genetic science. Common uses of bioinformatics embody the identification of candidate genes and nucleotides (SNPs).

Often, such identification is created with the aim of higher understanding the genetic basis of sickness, distinctive variations, fascinating properties (esp. in agricultural species), or variations between populations. During a less formal means, bioinformatics additionally tries to know the structure principles inside macromolecule and super molecule sequences, referred to as genetics [1].

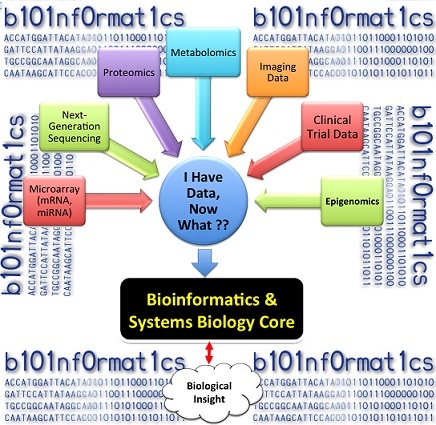
Bioinformatics could be a cognitive content field primarily involving biology and biological science science, technology, arithmetic, and statistics. Info intensive, large-scale biological problems area unit addressed from a method purpose of browse. The foremost common problems area unit modeling biological processes at the molecular level and making inferences from collected info. A bioinformatics answer generally involves the following steps:

• Collect statistics from biological info.

• Build a method model.

• Solve a method modeling draw back.

• Test and assess a method rule.



**Figure 1.1:-** Bioinformatics and System Biology.

Sequence alignments constitute amazingly powerful methods for uncovering the constraints forced by the structure and function on the development of a protein and nucleic acid family, this alignment task of multiple sequences requires vast measure of computational time.

It incorporates the task of finding and separating a couple of regions from two given biological sequences that show high similarity. Global alignment is additionally in light of completeness and is done over the whole sequence length to include as whatever number matches as could reasonably be expected up to and including sequence end whereas Pair wise Sequence Alignment is utilized to recognize regions of similarity that may indicate structural, functional or evolutionary relationships between two biological sequences (protein or nucleic acid).

This kind of alignment is basically based on numbers, multiple sequence alignment (MSA) is the alignment of at least three or more biological sequences of similar length and in this manner, it is included in the alignment based on numbers. From the output of MSA applications, homology can be surmised and the developmental relationship between the sequences can be considered.

* 1. **Open Research Issues**

1. Current MSA programs for deoxyribonucleic acid sequences usually model ball-shaped domain structure and evolution. Notwithstanding, deoxyribonucleic acid are unstructured (natively disordered) or contain giant unstructured regions.
2. From previous few year quantity of labor are wiped out the sphere of deoxyribonucleic acid sequence alignment however, still need a lot of work for the less time consumption in sequencing.
3. The result of the new sequences ensuing from high output Biotechnologies; we have a tendency to known sequence discrepancies which may result to fragmental or inaccurate sequences exploitation associate empirical rule-based approach.
4. New high throughout biotechnologies are providing North American country with enough information to create complete organic process histories of enormous sets of genes [10].
5. New technologies manufacture browse lengths as short as 35–40 nucleotides, leading to fragmental macromolecule sequences that cause issues for bioinformatics analyses [11].
6. Dependability is that the most complicated idea within the deoxyribonucleic acid sequencing which is influenced by assumption, disposition expectation, behavior setting and alternative problems. Many reliable Algorithms are developed for deoxyribonucleic acid sequence however as a result of synchronic increase and reduce of dependability and value severally generation of best resolution is extremely tedious task.
7. The macromolecule families used as benchmark takes a look at sets were chosen to produce a spread of various multiple alignment issues. Thus, the amount of sequences in every alignment ranges from four to 807. The mean sequence length for associate alignment ranges from fifty six to 3271 and mean residue % identity ranges from eleven to sixty eight.
   1. **Motivation**

The main objective of this thesis is to conduct associate empirical analysis of deoxyribonucleic acid sequencing as a replacement fragmented alignment approach to finding the MSA drawback. During this analysis, the pertinence of the ACO algorithms as a triple-crown sequence positioning optimizer is investigated.

Many alternative techniques, like T-COFFEE [84] and CLUSTAL X [112], are compared with ACO. Blessings and downsides of those strategies are mentioned, and ways in which to boost MSA exploitation ACO are explored. In fact, ACO algorithms have not been applied in the manner delineate during this thesis. This thesis so represents a number one experiment of its kind. The first objectives of this thesis area unit summarized as follows:

* To give Associate in nursing familiarity with the conception of biological sequence alignment, with a stress on MSAs.
* To give a summary of the relevant alignment techniques that may function valuable candidates for finding MSA issues.
* To study the consequences and performance of optimizing totally different sequence alignment objective functions (scoring schemes).
* To conduct a measurability analysis by perceptive the impact of ACO on MSA issues that differs by their complexness (number of sequences within the set, length of sequences and overall similarity).
* To investigate the performance of ACO, as applied to the MSA drawback.
* To compare solutions from ACO with solutions from alternative MSA programs so as to position ACO performance with relevance many factors, like alignment accuracy, time complexness.
  1. **Problem Definition**

Multiple sequence alignment (MSA) is in essence a double objective optimization problem: Gaps ought to be inserted into the first sequences in such the simplest way that (3) the amount of matching characters is maximized and (4) the amount of gaps inserted is reduced. Bearing in mind that the 2 needs area unit conflicting, Associate in Nursing best alignment answer cannot continually be found. Several previous sequence alignment techniques are utilized, like tree-based algorithms [5] that mix results from combine wise alignments.

The most drawbacks with these algorithms are that they assume the existence of a tree that properly describes the relationships between sequences. As a result of such trees cannot continually be derived or calculated, a shortage still remains within the techniques out there to unravel the MSA drawback.

Most of the ordinarily used MSA strategies [8] area unit supported dynamic programming (DP) [9]. However, stateless person needs time and memory proportional to the merchandise of the sequence lengths. Hence, several heuristic strategies [6, 7] are developed to search out sensible alignments, that don't seem to be essentially best, among an affordable time.

* DNA sequence style downside may be a multi-objective improvement (MOO) drawback, where quite one objective should be optimized subjected to several constraints. Given several short stranded DNAs in associate degree passing tube, these DNAs tend to cross to various molecules inside the tube subject to Watson-Crick complement once the temperature is down. The deoxyribonucleic acid sequence vogue drawback is to avoid these hybridizations once the temperature is down. The possibility of a sequence to cross with itself and various DNAs is also lived H measure, similarity, hairpin, and continuity. These objectives unit subjected to rate content and melting temperature constraints. Generally, given style of objective functions and constraints in polymer sequence vogue, the target of the matter is to vogue and manufacture sets of fantastic polymer sequences with reduced values of the target functions. If this condition is achieved, it's going to be said that the sequences at intervals the set area unit distinctive and can't cross to every alternative.
  1. **The target of Thesis**

The aim the thesis is to style and implement the new framework for maintaining the objectives of so as to realize the first objective many intermediate objective area units outlined. These objectives area unit divided into 3 categories: preliminary, main and post objective

* + 1. **Preliminary Objective**

The preliminary objective is objectives that require to be consummated so as to achieve the sufficient information to meet the most objective.

* DNA sequence style downside may be a multi-objective improvement (MOO) downside, wherever quite one objective must be optimized subjected to many constraints.
* The main objective of this thesis is to conduct AN empirical analysis of ACO with new fragmented approach to finding the deoxyribonucleic acid sequence alignment downside.
* In this analysis, the pertinences of the ACO algorithms as a in sequence positioning optimizer is investigated.
* Several alternative techniques, like T-COFFEE [84] and CLUSTAL X [112], area unit compared with ACO.
* Advantages and downsides of those strategies area unit mentioned, and ways that to boost MSA exploitation ACO area unit explored.
* In fact, ACO algorithms haven't been applied within the manner delineated during this thesis. This thesis there for represent a number one experiment of its kind.

**1.5.2 Main Objectives**

The fulfillment of the most objectives results in the fulfillment of the first objective.

* To offer a familiarity with the construct of biological sequence alignment, with a stress on MSAs.
* To offer an summary of the relevant CI techniques that will function valuable candidates for finding MSA issues.
* To study the consequences and performance of optimizing completely different sequence alignment objective functions (scoring schemes).
* To conduct a quantifiability analysis by perceptive the impact of ACO on MSA issues that differs by their quality (number of sequences within the set, length of sequences and overall similarity).
* To investigate the performance of ACO as applied to the MSA drawback.

**1.5.3 Post Objective**

* This objective has to be determinant to what extend the first objective has been consummated and to spot potential areas of improvement.
* Simulate behavior of the enforced dependableness analysis square measure carried and analyze results. Simulations with the enforced extension have to be compelled to be administrated. The results should be analyzed to see the impact of dependableness and price over the configuration and selects the potential space which might be makeshift.
* We value our projected formula by examination this with exiting basic framework of Particle Swam optimization (PSO).
  1. **Contribution of the analysis Work**
* The first contribution of this work is that the new model accustomed vogue polymer sequences exploitation ACO formula. This model represents a dimension of the search house conjointly sequence. Therefore, a particle among the formula carries extra twelve than one sequence that would be a collection of polymer sequences. Different researchers have implemented ACO formula with altogether totally different illustration of model and search house.
* The second contribution of this work is that the event of ACO formula for polymer sequence vogue. This work is believed to be novel since there isn't any analysis work has been administrated to implement ACO to resolve the polymer sequence vogue downside.
* The third contribution throughout this work is that the employment of ACO methodology to resolve the sequence alignment drawback of deoxyribonucleic acid sequence style. During this methodology, four objectives square measure depicted exploitation four swarms.

Every swarm minimizes just one objective operate, and by the top of every iteration, these ants share their data with one another. They notice their method by secretion. it's conjointly found that to date no analysis has used like” fragmented deoxyribonucleic acid sequence alignment exploitation ACO” methodology for deoxyribonucleic acid sequence style.

* A novel application of ACO to the sector of bioinformatics addressing sequence alignment and conjointly to seek out the shortest path.
* The derivation of 2 illustration schemes for the MSA drawback, particularly fragmented technique and deoxyribonucleic acid.
* The introduction of a gap-reducing issue supported overall similarity so as to lower the quality in MSAs.
* The development and analysis of latest ways that of optimizing sequence alignment objective functions. The list below presents the organization of the chapters which make up this thesis. Also Given is a brief description of the topics each chapter deals with.
  1. **Thesis Layout**

The list below presents the organization of the chapters which make up this thesis. Also Given is a brief description of the topics each chapter deals with.

* **Chapter** [**2**](#page19) coversthe necessary background relating to methods designed to alignDNA sequences. This chapter includes a theoretical review of DNA, sequence alignment, Bioinformatics, python and current alignment methods.
* **Chapter** [**3**](#page44) it covers literature review. This review describes about the related work in the given topic in the field on sequence alignment algorithm and swarm intelligence techniques followed by different issues, application areas and also the literature review explores currently using processing techniques providesa brief survey.
* **Chapter 4** It’s covers the problem statement and also includes the proposed method with its proper algorithm and its description.It’s provides the details of the implementation environment, what modules and tools have been used and also cover the technologies involved in it.. Thecore of this thesis resides in this chapter. All aspects relating to the experimental procedure are covered, comprising mainly the experimental methodology, description of experiments, experiments themselves and analysis of the results.
* **Chapter 5** It’s includes results and analysis where all the results obtained from the tool have been plotted and then analyzed in order to conclude the overall research work. Providesa summary of the findings of this thesis, and considers possiblefuture research that emanates from it.

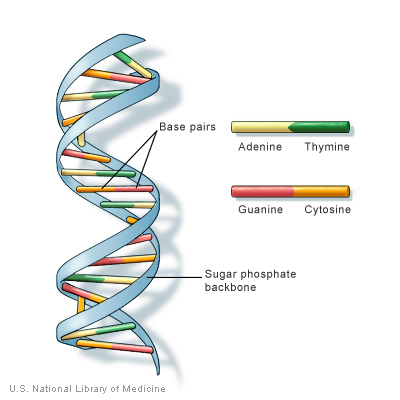
The following appendices are also included. They contain supplementary material related to the main text of this thesis, as well as a number of lists containing relevant information for quick referencing purposes.

**CHAPTER -2**

**THEORETICAL BACKGROUND**

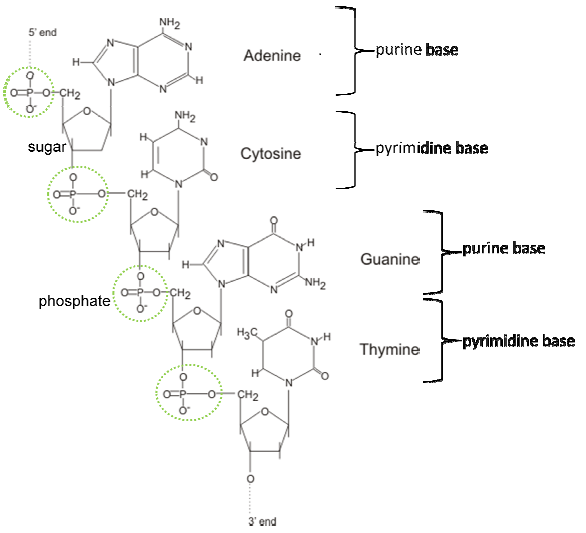
**2.1 DNA**

Deoxyribonucleic acid (DNA) may be a brilliant molecule that contains the genetic directions utilized within the event and functioning of all acquainted living organisms and some viruses. The foremost role of deoxyribonucleic acid molecules in living organism is that the long-run storage of information. Chemically, DNA may well be a compound that's connected on from a series of monomers. Monomers that kind the structure of nucleic acids, unit called nucleotides. Each organic compound contains a sugar (deoxyribose), a phosphate cluster, and one kind every of four bases: A (A), T (T), G (G), or C (C). These bases square measure classified into 2. A and G square measure known as prune bases as a result of their structure consists of two rings of atoms. On the other hand, C and T unit known as rig bases, since they have one ring of atoms. Each base incorporates a rather utterly completely different composition, or combination of gas, carbon, nitrogen, and number one.



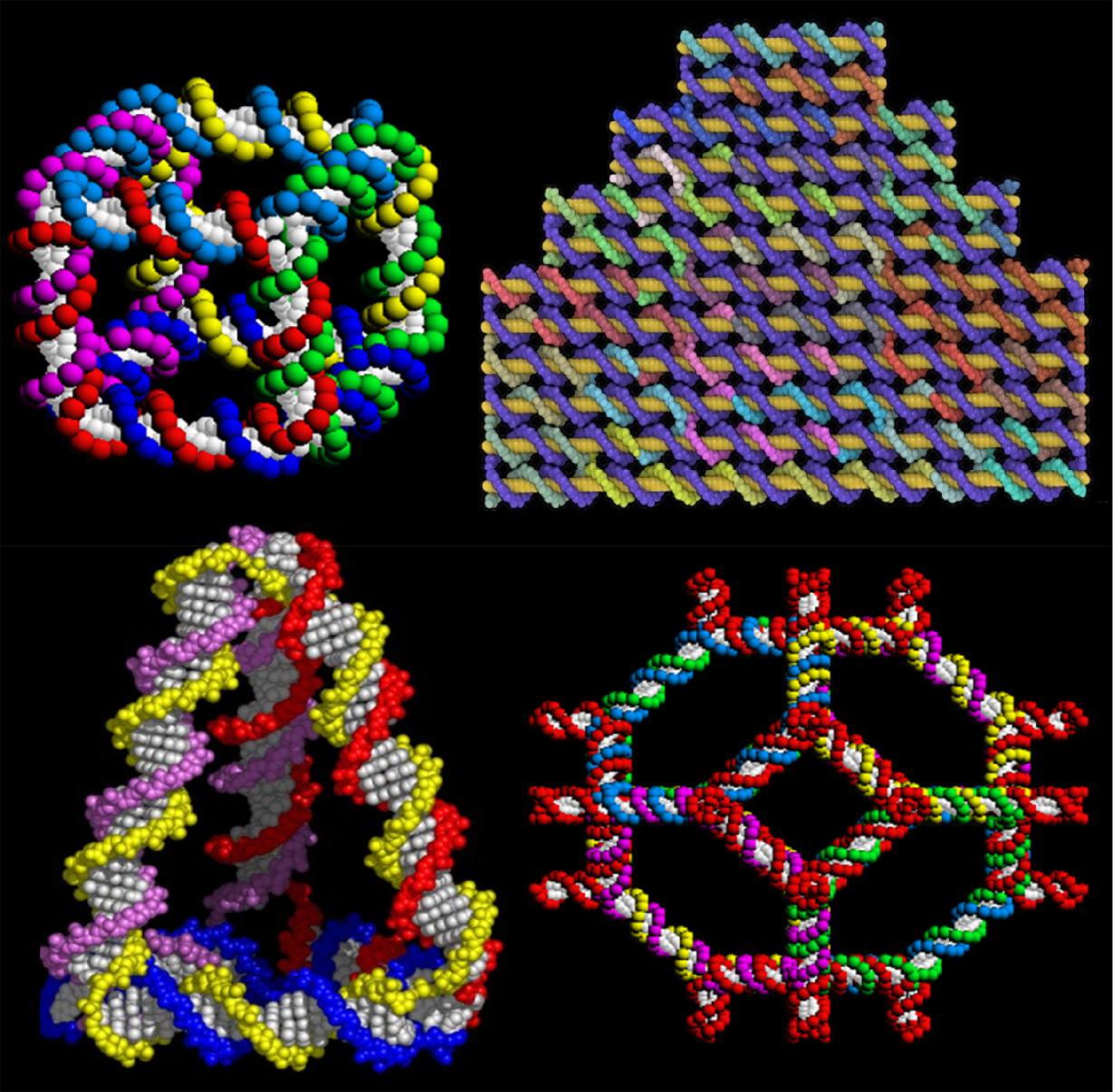
**Figure 2.1:-** Structure of DNA with Amino acids.

A fiber deoxyribonucleic acid consists of a series of nucleotides that incorporates a sense of direction, within which one finish is with chemicals completely different than the opposite. The two fiber deoxyribonucleic acid square measure control along by atomic number 1 bonds between pairs of bases, that square measure known as duplex or double-stranded polymer supported Watson-Crick complement. Each type of base on one strand forms a bond with just one type of base on the other strand. The nucleotides alone kind stable bonds in certain combinations: A pairs with T, and G pairs with C, as shown in Figure one.2 (Seiffert and Huhle, 2008). Thus, A-T and G-C base pair’s square measure aforementioned to be complementary. As shown in Figure one.2, purines type atomic number 1 bonds to pyrimidines, with A bonding exclusively to T, and C bonding exclusively to G. This arrangement of two esters binding on across the spiral is called a nucleotide.



**Figure 2.2:-** Chemical structure of each base and phosphate in single-stranded DNA.

Hybridization could be a technique or method that has been found by chemist Roy conductor within the Nineteen Sixties as the way to analyses the composition of ordering. By understanding conjugation, the tactic that single DNA strands combine to create a helix is prime to biology and central to DNA-based technologies like DNA computing, DNA biotechnology, And DNA field of study. Denaturation can be a technique where double-stranded DNA strands uncoils and separate into fiber DNAs, whereas conjugation happens once DNA strands bind, or crossbreed supported Watson-Crick base pairing. In denaturation, double-stranded DNAs are



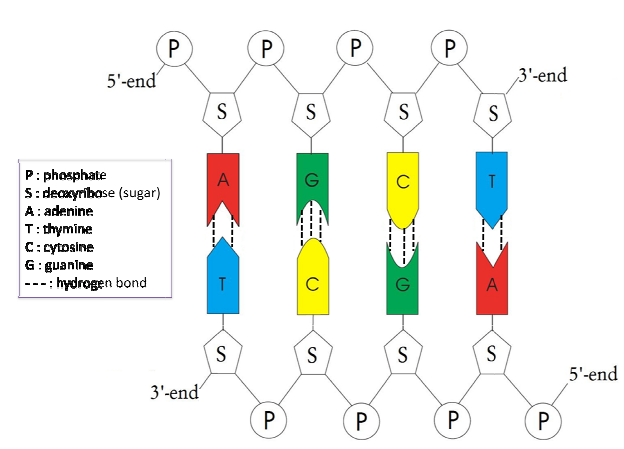
**Figure 2.3**: Examples of DNA Nanostructures (Seeman*et al.*, 1998).

going to be separated by heating up the solution to relating to 85-95°C, whereas conjugation are going to be done by cooling down the tube reaction answer [12]. There are a unit three kinds of conjugation, significantly bi-molecular conjugation, multi-molecular conjugation, and Uni-molecular conjugation. Bi-molecular conjugation happens once a pair of kinds of fiber DNAs kind a helix structure of DNA as shown in Figure one.3. Meanwhile, 3 or a lot of strands square measure concerned within the multi-molecular hybridizing, that is that the essence of Adleman deoxyribonucleic acid computing [13].

Uni-molecular cross or self-hybridization could develop a pin formation as shown in Figure one.4. This might happen if a complementary subsequence exists among an equivalent fiber deoxyribonucleic acid.

Deoxyribonucleic acid cross is extraordinarily sensitive to deoxyribonucleic acid sequence or composition. Knowledge of but the method happens may modify researchers to a lot of strategically style technologies.

For instance, if a research worker needs to style sequences that bind terribly quickly or with high potency, he or she may place sure bases in specific locations, so the hybridizing reaction may occur quicker or a lot of faithfully.

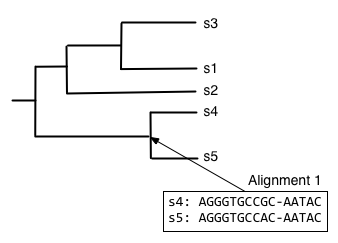


**Figure 2.4:-** A double-stranded DNA structure.

**2.2. Sequence Alignment**

In bioinformatics, a sequence alignment is also a way of arrangement the sequences of deoxyribonucleic acid, RNA, or super molecule to spot regions of similarity that may be a consequence of sensible, structural, or process relationships between the sequences.

Aligned sequences of ester or compound residues unit typically pictured as rows at intervals a matrix. Gaps unit inserted between the residues thus identical or similar characters unit aligned in sequent columns. Sequence alignments also are used for non-biological sequences, like hard the edit distance price between strings in an exceedingly tongue or in money knowledge.



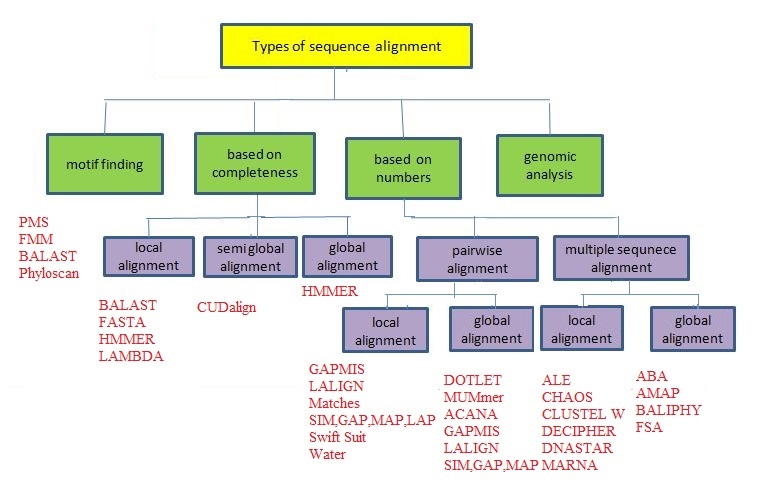
**Figure 2.5:-** Alignment of five Sequences.

**2.2.1. Pair wise alignment**

Pair wise sequence alignment strategies area unit accustomed realize the best-matching piecewise (local) or international alignments of two question sequences. Combine wise alignments can exclusively be used between a pair of sequences at a time, but they unit of measurement economical to calculate and are generally used for ways that do not would like extreme accuracy (such as trying a data for sequences with high similarity to a query).

The three primary ways of producing combine wise alignments unit of measurement dot-matrix ways, dynamic programming, and word methods; [1] but, multiple sequence alignment techniques can also align pairs of sequences. Each technique has its individual strengths and weaknesses, all three combine wise ways have issue with very repetitive sequences of low knowledge content - notably where the number of repetitions differ inside the two sequences to be aligned.

Some way of quantifying the utility of a given combine wise alignment is that the 'maximum distinctive match' (MUM), or the longest subsequence that happens in every question sequences. Longer MUM sequences generally mirror nearer connectedness.



**Figure 2.6:-** Hierarchy of sequence Alignment Algorithms.

**2.2.2. Dynamic programming**

The technique of dynamic programming is also applied to produce international alignments via the Needleman-Wunsch formula, and native alignments via the Smith-Waterman formula. In typical usage, super molecule alignments use a substitution matrix to assign scores to amino-acid matches or mismatches, associated a distinct segment penalty for matching associate chemical compound in one sequence to a distinct segment inside the various.

DNA and compound alignments may use a rating matrix, but in observe typically simply assign a positive match score, a negative match score, and a negative gap penalty. a typical extension to plain linear gap costs, is that the Dynamic programming is also useful in positioning organic compound to super molecule sequences, a task tough by the need to want into thought frame shift mutations (usually insertions or deletions).

**2.2.3. Multiple sequence alignment**

Multiple sequence alignment is Associate in nursing extension of mix wise alignment to incorporate quite two sequences at a time. Multiple alignment ways in which try to align all of the sequences throughout a given question set.

Multiple alignments are sometimes utilized in characteristic preserved sequence regions across a bunch of sequences hypothesized to be evolutionarily connected. Such preserved sequence motifs are going to be utilized in conjunction with structural and mechanistic information to search out the action active sites of enzymes.

Alignments are accustomed aid in establishing process relationships by constructing process trees. Multiple sequence alignments are computationally robust to provide and most formulations of the matter cause NP-complete combinatorial optimization problems [7][8].

**2.2.4. Progressive ways**

Progressive, hierarchic, or tree ways in which generate a multiple sequence alignment by initial oriented the foremost similar sequences and then adding successively less connected sequences or groups to the alignment until the whole question set has been incorporated into the solution.

The initial tree describing the sequence connation depends on try wise comparisons which is able to embrace heuristic try wise alignment ways in which rather like FASTA. Progressive alignment results area unit hooked in to the choice of "most related" sequences and then are going to be sensitive to inaccuracies among the initial try wise alignments.

Most progressive multiple sequence alignment ways in which in addition weight the sequences among the question set keep with their conation that reduces the prospect of making a poor choice of initial sequences and then improves alignment accuracy.

**2.2.5. Repetitive ways**

Iterative ways commit to improve on the intense dependence on the accuracy of the initial mix wise alignments that's that the liability of the progressive ways in which. Repetitive ways in which optimize Associate in Nursing objective perform supported a selected alignment marking methodology by assignment Associate in Nursing initial world alignment so realigning sequence subsets.

The realigned subsets square measure then themselves aligned to produce consecutive iteration's multiple sequence alignment. Varied ways in which during which of selecting the sequence subgroups and objective perform square measure reviewed in. [14]

**2.2.6. Motif finding**

Motif finding, together known as profile analysis, constructs world multiple sequence alignments that conceive to align short preserved sequence motifs among the sequences at intervals the question set. Usually this can be often usually done by initial constructing a general world multiple sequence alignment, once that the very preserved regions square measure isolated and used to construct a set of profile matrices. The profile matrix for each preserved region is organized variety of a marking matrix but its frequency counts for each amino organic compound acid or organic compound at each position square measure derived from the preserved region's character distribution rather than from a further general empirical distribution.

**2.2.7. Structural alignment**

Structural alignments, that are generally specific to super molecule and generally chemical compound sequences, use information regarding the secondary and tertiary structure of the super molecule or chemical compound molecule to assist in orienting the sequences. This method is employed for two or extra sequences and generally manufacture native alignments; but, as a result of they rely upon the provision of structural data, they'll solely be used for sequences whose corresponding structures area unit best-known (usually through X-ray physical science or magnetic resonance spectroscopy). Structural alignments area unit used as a result of the "gold standard" in evaluating alignments for homology-based super molecule structure prediction [17] as a result of the expressly align regions of the super molecule sequence that area unit structurally similar instead of relying completely on sequence data.

## **2.3 Bioinformatics**

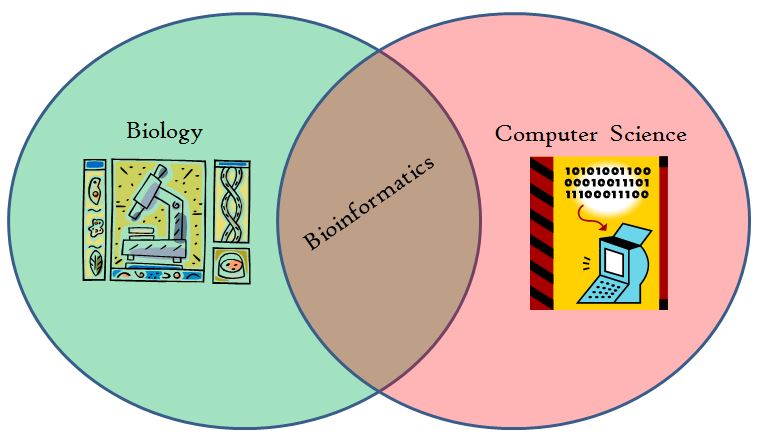
In the most up-to-date, a lot of biology-related info has collected than within the former over 2 thousand years of the history of science. This new surge of knowledge for the foremost half contains of super molecule and super molecule sequences mainly as a result of the means that DNA sequencing has became a standard procedure when the deoxyribonucleic acid revolution.

Bioinformatics has up as another field at the interface of science and biology within the mid-1980s, with the purpose of storing and analyzing the large quantity of knowledge gave by DNA sequencing. Bioinformatics consolidates numerical calculations algorithms, laptop science and measurements (i.e. science techniques) to induce learning from process examination of trial biological knowledge.

From a molecular biological viewpoint, for the foremost half manages the storage, retrieval, and analysis of super molecule and nucleic acid-derived organic compound sequences of proteins.

Its explicit subfields. For instance, structural bioinformatics manages the in silico analysis of the three-dimensional structure of macromolecules. Past sequencing, a vast live of knowledge is delivered by varied different supposed high-throughput (HTP) strategies that may be overseen simply by bioinformatics.

These HTP techniques incorporate, simply to mention many, organic phenomenon examination, ionophoresis and mass spectrographic analysis that make info to create up genetic, metabolic, signal transduction, protein-protein and different interaction pathways and systems.



**Figure2.7:-** Biology with Computer Science makes Bioinformatics.

Gives the middle tool compartment to the rising new field of systems biology. Systems biology plans to understand biology by AN all encompassing methodology and relies, additionally to different things, on the massive datasets provided by HTP ways.

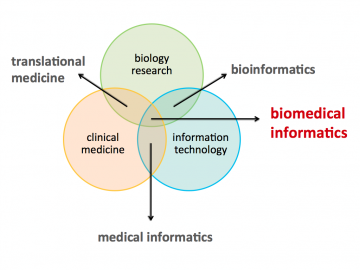
These methodologies grow the ancient theory approach of biology. The fields square measure a chunk of framework biology that started with genetics (Genome is that the full supplement of genetic material within a living being) followed by completely different fields of study, named utilizing non-standard speech language neologisms as genetic science (huge-scale study of the protein, the total complement of proteins at intervals AN organism), transcriptomics (the transcriptome may be a full supplement of deciphered RNA within AN cell or organism kind or a state of a selected cell), interactomics (study of the interactome, protein-protein associations within AN organism or cell).

One might proceed with AN „omics” summary to suppose the whole arrangement of little-molecule metabolites (metabolome), the whole arrangement of lipids (lipidome), the entire supplement of carbohydrates (glycome), the total arrangement of macromolecule enzyme enzymes (kinome) .

In this half, we'll depict the reputed essential databases that contain macromolecule and macromolecule sequences and in addition three-dimensional structures of macromolecules and their complexes. In addition, we'll provide and introduction to in silico sequence (and structure) analysis.

The part of bioinformatics in sub-atomic biological research tests (for example, restriction mapping of desoxyribo nucleic acid develops, coming up with of oligo ester primers) are secured in brief. The initial phases in sequence analysis square measure likeness searches and sequence alignments programs to project these analyses are delineated. Information from sequence alignments is used to create phyletic trees and to derive biological process connections among sequences (and among species).

In silico techniques are examined that square measure used to foresee structural and practical motifs at intervals macromolecule and macromolecule sequences. We must always bear in mind that the overwhelming majority of the sequence analysis information square measure predictions, and centre analyses ought to be directed to approve them.

. 

**Figure 2.8:-** Bioinformatics works with them.

**2.4 Python**

Python is also a general taken, interactive, object-oriented, and high-level linguistic communication. it had been created by Guido van Rossum throughout 1985- 1990. Like Perl, Python American Standard Code for Information Interchange document is in addition accessible beneath the bovid General Public License (GPL). This tutorial provides enough understanding on Python linguistic communication.

Python is also a high-level, taken, interactive and object-oriented scripting language. Python is supposed to be extraordinarily readable. It uses English keywords of times where as totally different languages use punctuation, and its fewer syntactical constructions than totally different languages.

• Python is interpreted: Python is processed at runtime by the interpreter. you're doing not ought to compile your program before execution it. typically|this can be} often like PERL and PHP.

• Python is Interactive: you will very sit at a Python prompt and move with the interpreter on to put in writing your programs.

• Python is Object-Oriented: Python supports Object-Oriented vogue or technique of programming that encapsulates code at intervals objects.

• Python is also a Beginner's Language: Python is also a pleasant language for the beginner-level programmers and supports the event of an outsized varies of applications from straightforward text method to computer network browsers to games.

**2.4.1 History of Python**

Python was developed by Guido van Rossum inside the late eighties and early nineties at the National analysis Institute for arithmetic and technology inside the country.

Python springs from many totally different languages, beside first rudiment, Modula-3, C, C++, Algol-68, Small Talk, and UNIX system shell and totally different scripting languages.

Python is proprietary. Like Perl, Python American Standard Code for Information Interchange document is presently accessible beneath the bovid General Public License (GPL).

Python is presently maintained by a core development team at the institute, the Guido van Rossum still holds a very important role in directional its progress.

2.4.2 Python choices

Python's choices include:

• Easy-to-learn: Python has few keywords, straightforward structure, and a clearly printed syntax. this allows the scholar to settle on up the language quickly.

• Easy-to-read: Python code is further clearly printed and visual to the eyes.

• Easy-to-maintain: Python's American Standard Code for Information Interchange document is fairly easy-to-maintain.

• A broad commonplace library: Python's bulk of the library is improbably transferrable and cross-platform compatible on UNIX system, Windows, and Macintosh.

• Interactive Mode: Python has support for AN interactive mode that allows interactive testing and debugging of snippets of code.

• Portable: Python can run on an outsized type of hardware platforms and incorporates a similar interface on all platforms.

• Extendable: you will add low-level modules to the Python interpreter. These modules alter programmers to feature to or customize their tools to be further economical.

• Databases: Python provides interfaces to any or all major business databases.

• graphical computer program Programming: Python supports GUI applications which will be created and ported to many system calls, libraries and windows systems, like Windows MFC, Macintosh, and additionally the X Window system of UNIX system.

• Scalable: Python provides a far higher structure and support for large programs than shell scripting.

Apart from the preceding choices, Python options an enormous list of fine choices, few sq. measures listed below:

• It supports purposeful and structured programming ways that additional as OOP.

• It is employed as a scripting language or is compiled to byte-code for building large applications.

• It provides really high-level dynamic data types and supports dynamic kind checking.

• IT supports automatic garbage pickup.

• It is just integrated with C, C++, COM, ActiveX, CORBA, and Java.

**CHAPTER -3**

**LITERATURE REVIEW AND SURVEY**

**3.1 Review of Sequence Alignment**

MSA may be a complicated drawback Associate in Nursing over the past four decades an increasing range of strategies are established that attempt to solve it, every with their own strengths and weaknesses. An outline of the properties and accessibility of the assorted strategies is mentioned as follows.

**3.1.1 BioPat**

BioPat may be a mathematical package that includes the first-ever integrated MSA technique that may be a world progressive algorithmic program with iteration talents [14]. Initially, a conjugation tree (dendrogram) is made supported all pair wise similarities of the sequences to be aligned, matched by dynamic programming.

The method provides decisions among several of the ordinarily used clustering techniques to make the dendrogram, such as Unweighted Pair-Group Mean Average (UPGMA), the present day relative technique or the Neighbor-Joining (NJ) technique. Once the dendrogram is completed, the sequences square measure more and more aligned following the branch order of the dendrogram..

**3.1.2 MULTAL**

The early technique MULTAL [15] is extremely quick and constructs a dendrogram throughout the progressive alignment, as within the technique of Feng and Jimmy Doolittle. It uses a quick serial branching technique to align the highest pairs of sequences initial and so later on align subsequent highest sequences to those already aligned.

The order during which the sequences square measure aligned is basically supported the worldwide organic compound composition of the sequences that saves the fixed charge of playing all-against-all pairwise alignments.

Blocks of aligned sequences square measure scored by dynamic programming the same as the tactic MultAlin, however the similarity of 2 alignment columns is to boot normalized by the minimum range of sequences in either of 2 compared alignment blocks.

**3.1.3 MultAlign**

The global progressive technique MultAlign [16] establishes a straightforward chain order during which the individual sequences square measure aligned one by one. Initially, all pair wise alignment scores square measure determined and also the 2 most similar sequences square measure matched initial. Throughout more iteration, the sequence showing the best alignment score once matched with the realigned sequence block is additional there to.

The MultAlign technique incorporates iteration capabilities in this the ensuing MSA will be more and more refined by realigning every sequence with the previous alignment from that that sequence is deleted i.e., sequence A1 is matched with aligned sequences A2...AN; sequence A2 is then realigned with the alignment of A1, A3...AN, so forth. This method is perennial till all N sequences square measure realigned.

**3.1.4 ClustalW, ClustalX**

ClustalW and also the later window graphic program (GUI) version ClustalX square measure the latest versions of the worldwide progressive alignment algorithmic program Clustal [17], and square measure typically thought-about because the normal technique for MSA. The progressive strategy used may be a simplification of the first Feng and Jimmy Doolittle them.

The alignment is made by initial building a guide dendrogram victimization Neighbor-Joining, supported sequence similarity, that is later on accustomed order sequential pair wise alignments.

The already aligned sequences square measure reduced to a profile for the next pair wise alignment. However, throughout the progressive alignment method, extremely specialized heuristics square measure applied to undertake and optimize however the sequence data is processed.

When the sequences are ordered for alignment in line with the pre computed dendrogram, the alignment of distantly connected sequences is delayed, so predominant the dendrogram. Conjointly the pair wise alignments are performed mistreatment native gap penalties and there's automatic selection and adjustment of the residue substitution matrix and gap penalties, severally. ClustalW and ClustalX perform best once the sequences to be aligned are global cases and don't have any obvious outlier.

**3.1.5 DCA, OMA**

The DCA (Divide-and-Conquer MSA) technique [18] is associate degree strict divide-and-conquer alignment formula. DCA follows an equivalent strategy because the MSA formula by Lipman and performs concurrent MSA rather than the progressive approach.

The DCA approach is associate degree challenge to beat the machine quality of the MSA technique. The divide-and-conquer strategy initial selects the longest sequence within the set to be aligned and cuts it near its point.

The remainders of the sequences also are cut at appropriate positions, which are calculated through a heuristic technique to scale back machine time, and consequently 2 new subsequence sets arise. This could then be perennial on the subsequence sets till an exact predefined minimum threshold for subsequence length is reached. The smaller the brink worth setting, the faster, however less optimum, the alignment becomes. The currently shorter sets of subsequences will then be singly aligned mistreatment the MSA formula, so reducing the time and memory needs.

At the end, all the sub alignments are concatenated to supply the total final alignment. DCA represents associate degree sweetening in accuracy and speed with regard to MSA, however machine time remains terribly sensitive to sequence distance and length so the amount of sequences that may be aligned still remains terribly low.

The DCA formula has conjointly been enforced as associate degree repetitious theme referred to as OMA.

**3.1.6 Dialign**

Dialign [19] may be a native consistency based alignment formula, which, rather than positioning single residues, aligns whole sequence segments. These segments may be imaginary as diagonals, as they'd seem on a dot plot of a matrix analysis.

The foremost recent version, Dialign2 [20] originally achieves all pair wise alignments of the sequences to be aligned, when that all ungapped segments (diagonals) are known. Consistent sets of diagonals are then determined associate degreed extra consecutive to the alignment mistreatment and repetitious mathematical procedure that determines the optimum order of addition.

Solely sequence fragments that matched segments are found are aligned; regions middle blocks of comparable segments are left unaligned. The development of Dialign2 compared to Dialign1 is that the modification of the initial weight of diagonals, that was form early based on Altschul and Erickson. The Dialign2 formula is each associate degree accuracy and machine time

**3.1.7 MEME**

The program culture [21] may be a tool for unattended motif looking among deoxyribonucleic acid and macromolecule sequences that operates mistreatment associate degree expectation maximization (EM) formula.

It discoveries occurrences of motifs by matching the residue compositional at every position of a purported motif against the overall composition of background sequence regions that don't show the motif. Regions viewing the foremost selective compositions are then selected as motifs.

A limitation of the culture motifs is that they're ungapped, however the program will realize multiple occurrences in individual sequences, that on the opposite hand don't got to be encountered among every input sequence.

Another helpful feature of the culture technique is that it is back-geared towards finding deoxyribonucleic acid word sequences, which are usually concerned as DNA binding sites for proteins.

**3.1.8 SAGA**

The program adventure story (Sequence Alignment by Genetic Algorithm) [22] is associate reiterative random alignment technique that uses a genetic algorithmic rule (GA) [23] to pick the alignment frogman evolving alignment population and that optimizes, as associate Objective perform (OF), the weighted total of pairs as employed in the MSA program.

The algorithmic rule primarily creates a random population of alignments of the sequences, referred to as generation zero (G0). Offspring alignments square measure then created from the parent alignments in G0 that square measure calculable for fitness supported alignment superiority.

The higher the alignment, the additional offspring alignments it produces. The operators for offspring alignment creation may be either the blending of the contents of the parent alignments (crossovers) or the alteration of one parent (mutation).

This method is iterated through sequent generations, permitting only the fittest (best-quality) offspring alignments to advance to consequent generation and turn out their own offspring alignments.

The iteration method halts once no additional improvement may be earned.

**3.1.9 T-Coffee**

T-Coffee (Tree-based Consistency Objective perform for alignment Evaluation) [24] may be an international progressive consistency-based algorithmic rule. Initially, all pair wise alignments of the sequences square measure performed twice: once with {the international worldwide alignment technique ClustalW wherever one global alignment is created, and once with the native alignment technique Lalign wherever ten top-scoring parallel native alignments square measure generated.

The results square measure pooled into a primary library of pooled weights for every non redundant residue combine. The pooled weight for every residue combine corresponds to the total of countless the world and native alignments containing that residue combine. Every alignment score is that the share sequence individuation of that alignment.

A library extension step is then done employing a procedure referred to as matrix extension to measure however residue pairs align with regard to alternative residues within the library, manufacturing triplet weights. These triplets square measure then accustomed assess however well sequences square measure aligned associated to the opposite sequences within the knowledge set, instead of gazing pairs of sequences in isolation.

However, T-Coffee has speed and process demand boundaries once alignments sequences of enormous sequences residues are achieved and should even fail to complete them on average-powered systems.

**3.1.10 MUMmer**

Genome-wide sequence alignments need very quick algorithms which will handle immeasurable nucleotides. The alignment system pantomimist [25] uses “suffix trees,” which permit for associate alignment of 2 entire genomes in linear time and house.

The program finds “maximal distinctive matches” (MUMs) between 2 input sequences. A suffix tree may be a distinctive character string wherever the sequences square measure equal. a replacement branch is made wherever they disagree.

A pantomime creates a suffix tree supported one (reference) sequence and streams the second (query) sequence against it. Actor has been wont to assemble contiguous from shotgun-sequencing to construct the whole order.

MAFFT the MAFFT program [26] is focused on the quick Fourier rework (FFT) for quick detection of homologous segments. Amino acids square measure denoted by volume and polarity values, yielding signal peaks if homologous segments square measure aligned. The recorded segments are amalgamated to a final alignment by dynamic programming.

**3.1.11 MUSCLE**

MUSCLE [27] is an extensively used program. it's earned the next rank in accuracy and a quicker speed compared to ClustalW and T-Coffee. It includes quick distance estimation victimization km per counting; progressive alignment employing a new profile performs referred to as the log-expectation score; and refinement victimization tree-dependent restricted partitioning.

MUSCLESMP [28] was the primary parallel try of MUSCLE on shared memory system. It achieves associate overall speeding of fifteen.2 on sixteen processors SMP system victimization Open MP. It absolutely was combined with the multithreaded formula in [29].

It used the bag of-tasks model. Tests on sixteen node cluster showed fascinating improvement for progressive alignments and potency scales with the advance within the drawback size. MUSCLE-based multi scale simulations [30] are conferred within the 2 forms of infrastructures: native HPC cluster and Amazon AWS cloud solutions.

It’s been joined with grid house virtual laboratory that allows users to develop and execute virtual experiments on the essential machine and storage resources through its web site primarily based interface.

**3.1.12 DIALIGN-TX**

DIALIGN-TX uses progressive and greedy approaches for segment-based MSA [31]. It integrated anchors optimizations for correct alignments. DIALIGN-TX-MPI it's the parallel version of DIALIGN-TX [32].

It uses associate reiterative heuristic methodology for MSA and produces alignments by concatenating ungapped regions with high similarity. it absolutely was enforced victimization each Open MP and MPI on a 28-cores heterogeneous cluster.

**3.1.13 MSA Probs**

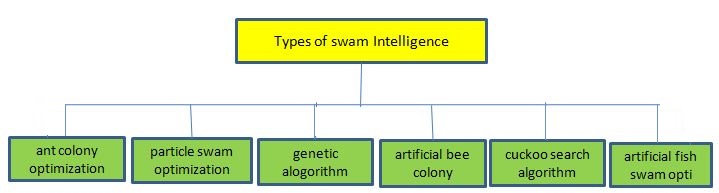
MSA Probs [34] could be a new and sensible multiple super molecule sequence alignment formula meant by combining a pair HMM and a partition perform to figure posterior chances.

It conjointly examines 2 essential bioinformatics techniques, particularly weighted probabilistic stability transformation and weighted profile-profile alignment, to attain high alignment accuracy.

Additionally, it's optimized for contemporary multi-core CPUs by using a multi-threaded style so as to decrease execution time. It statistically proves dramatic accuracy enhancements over many prime activity aligners.

**3.2 Literature Review of swarm Intelligence**

The area of metaheuristics in SI is chaotic because of the many “novel” metaheuristics that many of them are basically repeat ideas of existing metaheuristics [35] or they are not even inspired by nature or swarms, e.g., the fireworks algorithms inspired by the fireworks explosions [36]. Nevertheless, this paper focuses only on the SI metaheuristics applied to DOPs. ACO was developed for a discrete space whereas the remaining algorithms in Table 1 for a continuous space. The common characteristics of these algorithms are that they are inspired from nature, population-based, and iterative. Their differences, apart from their behavior inspiration, lie in the way the search space is explored and exploited by the “agents” [37].



**Figure 3.1:-** hierarchy of swarm intelligence (SI).

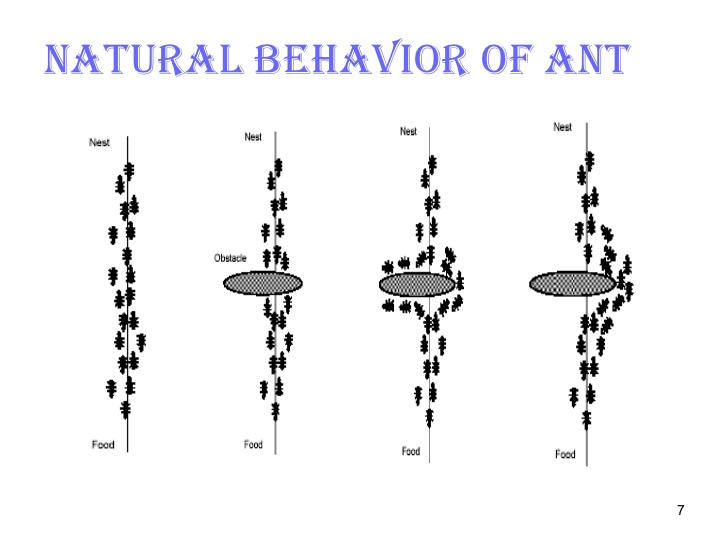
**3.1.1. Ant Colony Optimization (ACO)**

ACO is inspired by the foraging behavior of real ants. The goal of ants is to find the shortest path between their nest and food sources. ACO met heuristic is based on several construction steps and on a dynamic memory structure that contains information regarding the quality of previously obtained results [38, 39].

Each ant represents 1 a potential solution of the problem. ACO consists of a forward mode where ants construct their solutions probabilistically based on existing pheromone trails and heuristic information available a priori.

When all ants complete their forward mode they switch to their backward mode where a shared pheromone table is updated accordingly, i.e., the better the solution quality the more pheromone deposited. There are two main ACO frameworks, i.e., evaporation-based [40, 41] and population based [66x]. Their difference lies in the way pheromone is updated.

The evaporation based framework reduces the pheromone trails gradually by a constant amount to eliminate any previous poor old “decisions”. The population-based framework uses a population that removes pheromone trails directly when a solution is removed from the population.



**Figure 3.2:-** Natural Behavior of Ant.

**3.1.2. Particle Swarm Optimization (PSO)**

PSO was first introduced in [86x] to address continuous optimization problems. Each particle represents a potential solution of the problem. More precisely, each 270 particle consists of a velocity and position vectors, respectively, which are updated according to the best so far position of particle and the best so far position of the swarm.

There are two main models of the PSO algorithm, i.e., the global best and local best, respectively. Their difference lies in the neighborhood structure for each particle. In the global best model, the neighborhood of a particle consists of the particles in the whole swarm, which share information between each other.

On the contrary, in the local best model, the neighborhood of a particle is defined by several fixed particles. [42] stated that the global best model converges faster than the local best model whereas the former model has a higher probability of getting stuck in local 280 optima than the latter model. Surveys of different PSO variations can be found in [43, 44].

Both models are used but in different ways due to their characteristics. The global best model is normally used in multi-swarm based algorithms [45, 46, 47], while the local best model is commonly used in algorithms with a single swarm [48, 49, 50].



**Figure 3.3:-** Flock of swarms in Particle Swarm Optimization.

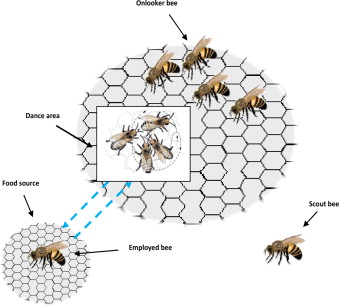
**3.1.3. Artificial Bee Colony (ABC)**

There are several developments of bee-inspired algorithms such as: ABC, bee colony optimization, bee system, marriage process bee, honey bee mating optimization, virtual bee algorithm, honey bee algorithm and beehive algorithm. Surveys of the different developments can be found in [51, 52, and 53].

The ABC algorithms that have attracted most of the attention, especially in DOPs [53]. In particular, an ABC algorithm mimics the behavior of real bee’s colonies [58]. A conventional ABC algorithm consists of food sources, whereas each food source represents a potential solution of the problem. Food sources are updated by three groups of bees: employed, onlooker and scout bees. Within the employed bee phase, bees search for new solutions. In particular, each bee produces a new candidate food source position from the old one.

In case food sources with more nectar are found, i.e., the new solutions have better fitness than the current, and then they are updated. Next, the relative probabilities according to the fitness determined from the employed bee phase are determined in the onlooker bee phase. Then, onlooker bees select a solution probabilistically in which the fittest solutions have a higher probability to be selected by onlooker bees.

After that, onlooker bees have the same behavior with the employed bees. Finally, scout bees randomly reallocate solutions in case they are abandoned, e.g., they have not been updated for a certain time.



**Figure 3.4:-** Behavior of Bee’s in Artificial Bee Colony (ABC)

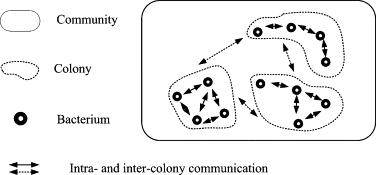
**3.1.4. Bacterial Foraging Optimization (BFO)**

The BFO algorithm is inspired by the complex organized activities in bacterial foraging and the survival of bacteria in different environments [59, 54, and 55]. A BFO algorithm consists of several bacteria, which represent solutions in the optimization problem and consists of three processes: chemo taxis, reproduction, and elimination dispersal.

In chemo taxis, a bacterium with random direction represents a tumble and a bacterium with the same direction of the previous step indicates a run. Next in the reproduction process all bacteria is sorted and only half of the fittest bacteria survive.

Then, the surviving bacteria are split into two identical ones to form the new bacteria. Finally, in the elimination-dispersion process, a bacterium is chosen probabilistically to move to a different random position in the search space.

Although this action maintains the diversity during execution, it may disturb the optimization process and therefore it is performed after several steps of the reproduction process.



**Figure 3.5:-** Bacterial Foraging Optimization.

**3.1.5. Artificial Fish Swarm Optimization (AFSO)**

There are several existing developments of fish-inspired algorithms. A detailed description of developments can be found in [56]. The AFSO inspired by the foraging behavior of real fish swarms in water world [62] which was applied for DOPs.

Within AFSO, each artificial fish looks for a position (solution) with more food source (better fitness) by performing three main behaviors: prey, swarm and follow.

The prey behavior is performed by an artificial fish without considering other swarm members. More precisely, a target position better than the current is considered randomly within the visual of the fish. The swarm behavior is a group behavior and is performed globally among all members of swarm as follows. Each artificial fish consists of a number of neighbors within its visual.

In case the central position of the visual field is better; then it moves towards the central position; otherwise, the prey behavior is performed again. Similarly, the follow behavior is performed, but instead of moving toward the central position, the artificial fish will move toward a better neighbor position within its visual.

Otherwise, the prey behavior is performed again. Basically, the prey behavior is performed when an artificial fish is not able to move to a better position when follow or swarm behavior are performed. In case the algorithm reaches stagnation behavior some artificial fishes are selected randomly from the whole artificial fish swarm and are set randomly. The best so far artificial fish position (i.e., solution) is recorded.



**Figure 3.6:-** Artificial Fish Swarm Optimization.

**3.1.6. Firefly Algorithm (FA)**

The FA was inspired by the flashing patterns and behavior of fireflies [61, 62]. An FA is based on three assumptions. All fireflies can be attracted by all other fireflies. The attractiveness of each firefly is proportional to the brightness of other fireflies.

The landscape of the problem determines the brightness of fireflies. Hence, a firefly that is less bright will move toward a brighter one. Otherwise, if a firefly is not able to locate a brighter firefly, it will move randomly.

Each firefly glows proportionally to its solution quality, which, together with its attractiveness, dictates how strong it attracts other members of the swarm. Adapting in Changing Environments Since all SI algorithms were initially designed for stationary optimization problems, they share a common property: convergence, i.e., they are designed to converge to the optimum quickly and precisely.

In contrast, DOPs require repeated optimization and tracking of the moving optimum. However, when an algorithm converges, its adaptation capabilities are lost due to the diversity lost issue.

Therefore, it is important to address the diversity lost issue by increasing/maintaining the diversity. However, it does not mean that a high level of diversity will lead to better performance [63, 57].

This is because too much randomization may disturb the optimization process. Another important aspect of SI algorithms to adapt well in DOPs is to promote the knowledge transfer.

Naturally, knowledge can be transferred from previously optimized environments using SI algorithms, e.g., via pheromone trails with ACO, via the food sources with ABC and AFSO, via the position of fireflies and bacteria with FA and BFO, respectively. However, it may not be enough to quickly recover when a dynamic change occurs.

On the other hand, if too much knowledge is transferred, it may start the optimization process close to a poor local optimum and get stuck there. Enhanced SI algorithms have proved to be powerful for different DOPs.

The main idea of enhancement strategies integrated to SI algorithms is to achieve a good balance for knowledge transfer and diversity maintenance. These two factors are also conflicting because if diversity is not maintained then the algorithm will not be very flexible to utilize any knowledge transferred.



**Figure 3.7:-** Firefly colony.

**3.1.7 Linear Longest Common Subsequence**

It is the most common algorithm used to find the subsequence occurring in both strings, but not necessarily contiguous, Overlapping Substructure property can be used to avoid re-computation kof same sub-problems by either using memoization ( or tabulation).

This algorithm has Time & Space Complexity of O ( m x n ) ~ O(L2)

Where, ‘m’ is length of 1st string & ‘n’ is length of 2nd string

‘L’ is length of sequences, L = m = n , in general case, both sequences have same length

**Note:** Assume m >= n : this will help in some sort of the Optimization in next approach

**3.1.8 Circular Longest Common Subsequence**

Linear LCS cannot be directly used for DNA, RNA or protein due to its inability to detect sub-sequences which can occur in circular fashion, in bio-informatics this is a common problem, which is being neglected over a long period of time, as other approaches are developed for MSA problem But CLCS can help determining similarity between sequences with fairly short sequence length, selected from the main sequences to be compared

Beside, having high computational complexity than iterative methods, CLCS can determine similarity between sequences in reasonable amount of time, which cannot be achieved using Simple LCS

Iterative algorithms till date are using Simple LCS as their sub-routine, while we have used CLCS as sub-routine in ACO approach, to improve results generated by ACO, which will then be compared with FTLPSO with no such modification made, to retain idea last developed by author of base paper

Finding Circular LCS of two strings is even more complicated, as by naïve approach it will require O( m x n x n ) ~ O(L3) time, which is not a reasonable time for protein sequences, even if done pair wise.

Earlier, some authors have made attempt to develop an O( m x n ), algorithm for Circular LCS .

Below is an Example of 2 strings, which are same when considered in circular fashion, like simple proteins, & nature of both Linear & Circular LCS is compared:

Seq\_1 = “abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ“

Seq\_2 = “ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz”

Length of Simple LCS is 26, While Length of Circular LCS is 52

Finding Simple LCS takes 0.009433 Seconds

Finding Circular LCS takes 0.010010 Seconds

I have developed successfully a simple yet powerful algorithm to find out Circular LCS, in the same Time & Space Complexity of O ( m x n ) ~ O(L2), actually by multiplying by a factor of 2 in worst case, which doesn’t affect the runtime of both algorithms, as much as, it has impact on results, CLCS is introduced as it is to be used as sub-routine in next two iterative algorithms for comparative studies.

**CHAPTER -4**

**PROBLEM DEFINITION & PROPOSED SOLUTION**

* 1. **PROBLEM DEFINITION**

The present design, approach, implementation and evaluation for DNA sequence Alignment is multiple sequence alignment (MSA) is in essence a multi objective optimization problem: Gaps ought to be inserted into the first sequences in such the simplest way that the amount of matching characters is maximized and the amount of gaps inserted is reduced Literature survey & Base paper suggested the use of FTLPSO over other pre-existing algorithms for MSA problem

After careful analysis of both Fragmentation technique & Two-Layer PSO, we have figured out problem in K-Tuple approach which is also attempted to be rectified by us, which results in improvement of performance

Following are modification in K-Tuple algorithm:

1. Standard K-Tuple takes 1st sequence as a Query to generate index table for generating fragments,
2. When, we have attempted other Sequences to taken as Query, resulting index table generates more equally distributed fragments
3. K-Tuple is considered to be very fast algorithm, due to this reason, we have added some extra effort to it, which result in better index table
4. New k-Tuple approach taken each sequence as a Query at a time, & generates index table, which is then compared to previous stored table, if it is found to be better than previous table, currently generated table replaces the main table, & this process continues for all sequences, due to which best table comes as a output.
5. Each table is assigned a score, based on which uniformity of fragment sizes are taken into account into tables.
6. Score function if defined as:

Still, performance of resulting FTLPSO is not comparable to any variant of ACO method

Three variants of ACO are studied, comprising of standard ACO, ACO-S and Proposed ACO-C approach.

The main problem is with using PSO (particle swarm optimization) algorithm is it is easy to fall in local optimum in high dimensional space and has a low convergence rate in the iterative process. It start with a group of a randomly generated population, it does not guarantee success. Information sharing mechanism in PSO is significantly different and little bit difficult to other optimization techniques.

In particle swarm optimization topology is constant and too slow compared to classical approaches. Particle swarm optimization easily suffers from the partial optimism, which causes the less exact at the regulation of its speed and the direction. Particle swarm optimization cannot workout the problem of scattering and non-co-ordinate system, such as solution to the energy field and moving rules of particles in the energy field. It does not work well on long sequences.

**4.2. Introduction to proposed approach**

Based on the observations made upon the previous technique & its drawbacks, our work presents scalable ACO algorithm with some suggested modifications, which results in better values of objective functions taken into account for MSA problem.

Ant-Colony-Optimization is meta-heuristic inspired from Ants, actually ants are unable to see form longer distances, so they drop a chemical known as “Pheromone” continuously on the path they travel, which also get evaporated over time

In this way ants perform, “Stigmergic communication” (communication by changing environment). This approach taken by ants, can be used to solve various **NP-Complete** problems like TSP.

Environment in an Ant Colony changes by two means:

1. Each ant drops a constant amount of Pheromone
2. There is a constant Evaporation of fixed fraction from available Pheromone

**4.2.1 Motivation behind choosing ACO**

1. ACO method can directly work on longer sequences; hence it does not require explicit fragmentation to be done.
2. Due to above reason, performance drawbacks introduced by fragmentation technique are not present in work done by ACO, unlike previous TLPSO approach
3. ACO is highly scalable algorithm, & can be developed to work in parallel, hence utilizing maximum available CPU resources, present on modern Multi-core architectures. This is because, behavior & actions of each ant, is independent of what, other ants are doing at the same time
4. ACO can be developed to work on any model of Shared Memory parallel programming, popular variant of which are Open Mp for Multi-Core & Open CL & CUDA for development for GPU architectures which are highly parallel in nature.

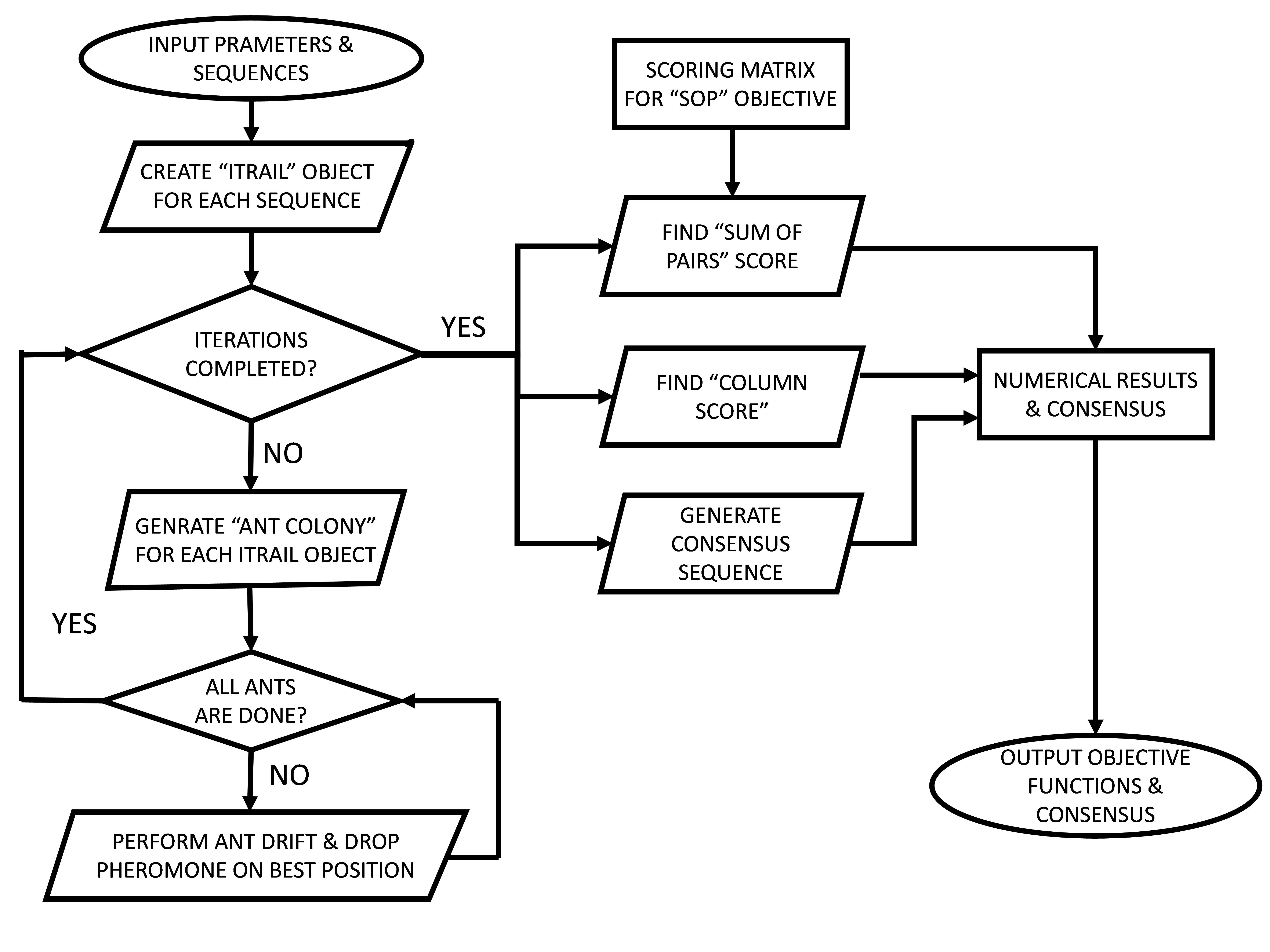
**4.3 Description of Proposed Approach**

The main idea of the system is that ants take a subsequence and move in an interval associated with each sequence, strengthening a pheromone trail when a close match (description of close match is the criteria, which can result in different variants of ACO) is found to a sequence at that position in other sequences. As the algorithm runs larger fragments of sequence are picked up by the ants.

Each sequence is assigned to an “ITrail” object in the system, which is responsible to maintain the pheromone trail associated with each sequence, & help determining in the evolved Consensus Sequences

Some objects of “Ant” class are assigned to each ITrail object, at random positions, each Ant object knows its current position & knows possible allowed drift range, in which it can move & drop pheromone at the position of all other ITrails, where it will find a close match, to the sequence associated to other ITrail object.

**4.4 Flowchart of Proposed Approach**

****

**Figure: - 4.1** Flowchart of proposed work.

**4.5 PSEUDO-CODE of Proposed Approach**

**Inputs:** Following are the inputs which is need to calculate sequence alignment of DNA.

1. **NS** as Number of Sequences to be aligned
2. **LN** as length of each sequence
3. **NC** as Number of Iterations
4. **AC** as Number of ants assigned to each ITrail objects
5. **SL** as Starting length taken by each ant on first iteration
6. **EL** as Ending length taken by each ant on last iteration
7. **DD** as Drift distance, which is maximum allowable drift of each ant
8. **RND** as probability of each ant dropping pheromone on a random position
9. **EVAP** as percentage of Rate of Evaporation after each iteration
10. **INTENSITY** as constant amount of pheromone which ant drops at any position

class **ITrail**

**{**

// ITrail class to maintain Trail & associated sequence, & perform evaporation after each iteration

def **constructor**( const , seq )

**{**

//”self” is the variable which refers to current object

//const is the constant amount to which, each trail is initialised

//Initialising pheromone trail

self.trail = numpy.array( [const]\*LN )

//Assigning sequence to object

self.seq = seq

**}**

def **evap**()

**{** //function, which perform evaporation of pheromone

self.trail = self.trail \* (1-EVAP/100)

**}**

**}**

class **Ant**

**{**

//Each Ant is associated with a ITrail object

def **constructor**( itrail , length , position)

**{**

self.length = length

self.pos = position

self.seq = trail.seq

self.itrail = itrail

**}**

def **make\_drift**( other\_itrail )

**{**

initialise ”all\_positions” to [ self.pos-DD, self.pos+DD ]

find best match of current sequence from “all\_positions” in other\_itrail

Let, that best match position be k

**if** ( random number in [0,1] < RND )

**{**

Drop pheromone at any position from “all\_positions” on trail of other\_itrail object

**}**

**else**

**{**

Drop pheromone at position k on trail of other\_itrail object

**}**

**}**

ITrail\_Objects = [ Itrail( 0,mat[i] ) for( i=0 ; i<NS ; i++) ]

rate\_of\_length\_increase = ( EL – SL ) / ( NC – 1 )

for ( c=0 ; c<NC ; c++ )

{

length = int( round( SL + rate\_of\_length\_increase \* I ) )

valid\_pos = [x for x in range(LN-length+1)]

generate “Any\_Colony” containing AC Ant objects for each of NS sequences

for( i=0 ; i<NS ; i++)

**{**

for( j=0 ; j<NS ; j++ )

**{**

if ( i is not equal to j )

for( k=0 ; k<AC ; k++ )

Ant\_Colony[ I ][ k ] . make\_drift ( ITrail\_Objects[ j ] )

**}**

**}**

for( i=0 ; i<NS ; i++)

Itrail obj[ I ] . evap ( )

**}**

Generate Consensus sequence, from the ITrail \_Objects

Evaluate both objective:

1. Column Score
2. Sum\_Of\_Pairs

**Output:**

1. Column Score Achieved
2. Sum\_Of\_Pairs Score Achieved
3. Length of Consensus Obtained
4. Derived Consensus Sequence

## CHAPTER -5

**SIMULATION & RESULT ANALYSIS**

**5.1 Simulation Tool**

We have developed an application to perform “Multiple-Sequence-Alignment” & find the so called resulting “Consensus” Sequence in Python3 programming language.

Python is a high-level programming language, which means it is possible to write codes in python, which can be executed in architecture independent fashion on any machine.

Python is purely interpreted language, which means each statement, which is verified by the syntactic guidelines of python is then translated into machine instruction being executed by Interpreter.

Python in Open Source, due to which, it undergoes a lot of bug fixes & upgrades time-to-time, mainly development is led by “Guido Van Rossum” who is also founder of Python.

Python is a Multi-Paradigm Programming languages, which means it provides support for k sequential, procedural, object-oriented & functional programming Python is most popular programming language of year 2016 & 2016

Python has few keywords, easy structure, and a clearly outlined syntax. This permits the coed to choose up the language quickly. Python code is additional clearly outlined and visual to the eyes. Python's ASCII text file is fairly easy-to-maintain.

Python's bulk of the library is incredibly moveable and cross-platform compatible on UNIX system, Windows, and Macintosh. Interactive Mode: Python has support for AN interactive mode that permits interactive testing and debugging of snippets of code.

Python will run on a large style of hardware platforms and has a similar interface on all platforms. Extendable: you'll add low-level modules to the Python interpreter. These modules alter programmers to feature to or customize their tools to be additional economical.

Python provides interfaces to any or all major business databases. Graphical user interface Programming: Python supports GUI applications that may be created and ported to several system calls, libraries and windows systems, like Windows MFC, Macintosh, and also the X Window system of UNIX system.

Python provides a much better structure and support for big programs than shell scripting. Apart from the preceding options, Python features a massive list of fine options, few square measures listed below:

It supports purposeful and structured programming ways further as OOP. It is used as a scripting language or is compiled to byte-code for building massive applications.

It provides terribly high-level dynamic knowledge sorts and supports dynamic kind checking. IT supports automatic trash collection. It is simply integrated with C, C++, COM, ActiveX, CORBA, and Java.

**5.2 System Specification**

Here in our research we are experimentally going to provide a simulation of recruitment post violence detection system where query will work with different type of conditions related to past behavior of query and data processing system and will help to provide accurate result.

Here are the described result screen and detail discussion about the result. Various operations we can provide and can be applicable once we perform the authentication success on the proposed technique. Here we have demonstrated our work in various respects and observed the result and measure the results based on the experiment performance

* Microsoft Windows-10 Operation System
* Inter-I family processor of CPU Clock Frequency >= 2.0GHz (preferably Quad-Core)
* 8GB DDR3 RAM with Memory Clock Frequency of 1600MHz
* 500GB of Hard Drive
* Python 3.6 with Number Processing library NumPy 1.12.1

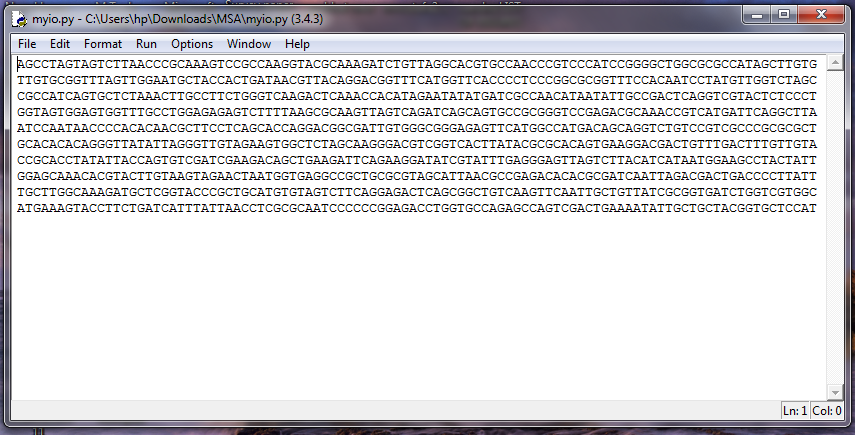
**5.3 Dataset**

Data file which can be simulated can be given as input and to process in the flow where the further distribution among the node can be perform.

A dataset is derived from the NCBI website. This is the website containing genome dataset in the entire requirement format. From this web a dataset taken by us is in .AA format and further index format is converted.

A Common dataset for Human DNA is collected from [www.biopython.org](http://www.biopython.org) which provides standard library for alignment & datasets to compare performance of various tools, with sliced length taken, to analyze the performance of techniques in short time durations

Biopython is a python package containing various tools to perform common operations done in bio-informatics, it is also open source & contain benchmarked datasets collected from various sites like <http://www.cs.utexas.edu>, <http://www-bio3d-igbmc.u-strasbg.fr/balibase/>,



**Figure:-5.1** data set which is used by proposed methodology.

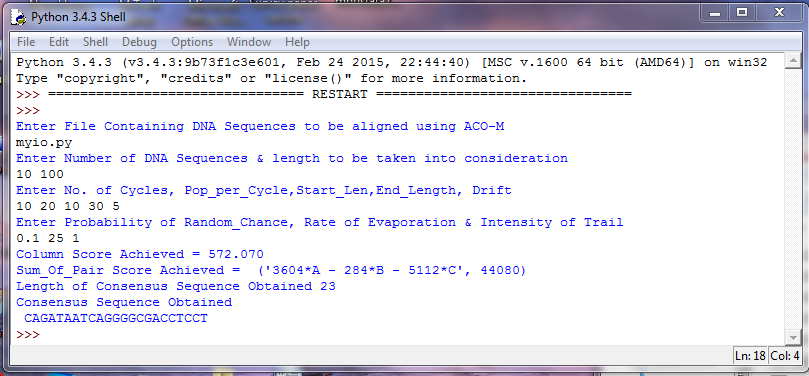
**5.4 Performance Analysis**

**5.4.1 Performance Analysis on same Dataset**

For iterative swarm intelligence based methods, Number of iterations & evolutionary elements taken per iteration are two measures able to clarify the performance of chosen method on dataset.

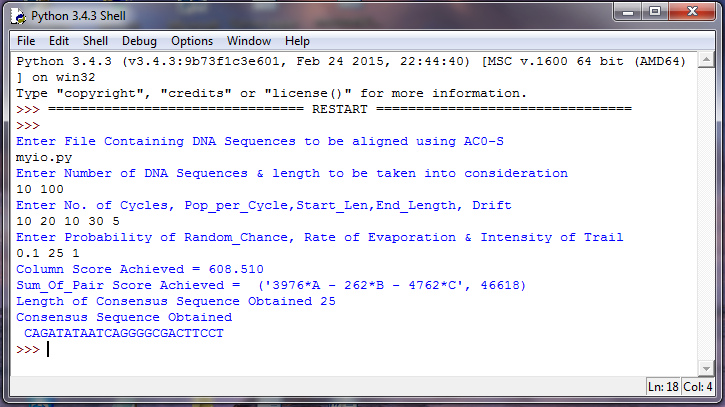
Swarm intelligence methods are massively scalable, so computational time is not generally taken as performance measure, which is taken into account when techniques like Dynamic programming & other classical algorithmic approaches are studied.

* **Ant colony Optimization with Pattern Matching Algorithm** (**ACO-M)**



**Figure:-5.2** Result of Ant colony Optimization with Pattern Matching Algorithm (ACO-M)

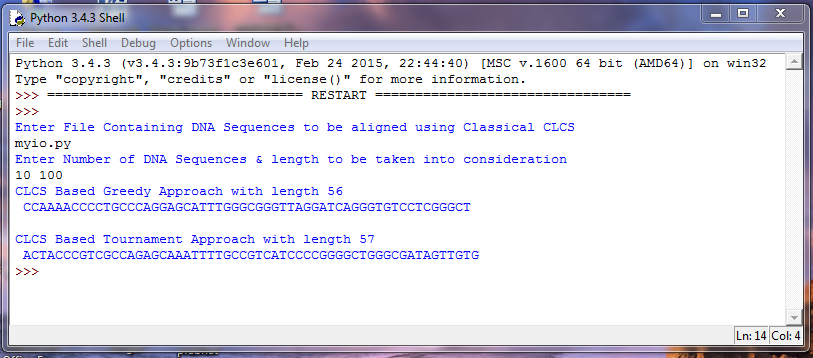
* **Colony Optimization with Longest Common Subsequence**(**ACO-S)**



**Figure:-5.3** result of Ant Colony Optimization with Longest Common Subsequence.

* **Cyclic Longest Common Subsequence (CLCS)**

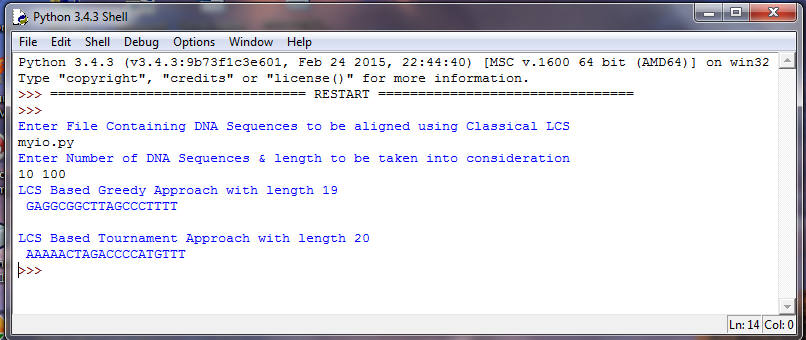
Linear LCS cannot be directly used for DNA, RNA or protein due to its inability to detect sub-sequences which can occur in circular fashion, in bio-informatics this is a common problem, which is being neglected over a long period of time, as other approaches are developed for MSA problem But CLCS can help determining similarity between sequences with fairly short sequence length, selected from the main sequences to be compared**.**



**Figure:-5.4** Cyclic Longest Common Subsequence (CLCS).

* **longest comman subsequence** (**LCS)**

It is the most common algorithm used to find the subsequence occurring in both strings, but not necessarily contiguous, Overlapping Substructure property can be used to avoid re-computation of same sub-problems by either using memorization (or tabulation).

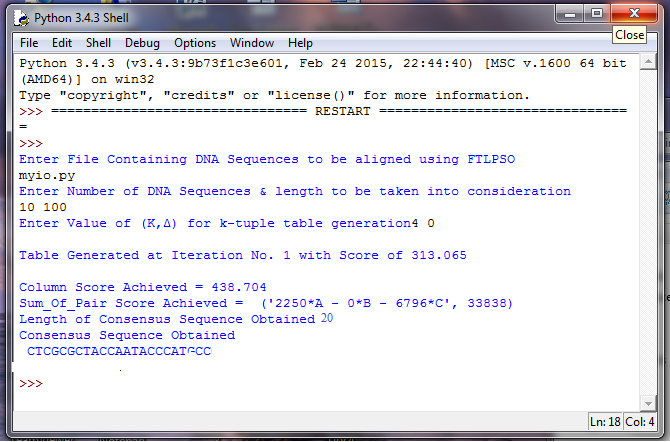


**Figure:-5.5** Result of longest comman subsequence (LCS).

* **FTLPSO**

This algorithm is divided into two main steps:

1. Fragmentation process, to shorten the longest sequences, makes working of PSO easier. K-tuple method is selected as a fragmentation technique & further modified to overcome its shortcoming, which can result in non-uniform fragments sizes. By the end of fragmentation process, an index table for fragment position is created similar to what is created using simple K-Tuple algorithm.
2. PSO does alignment process on fragments. Two layers of swarms are created, as the swarms in each layer will focus on one of the two scoring functions being taken (Column-Score & Sum-Of-Pair) of the MSA to appraise their performance. In each swarm, local-PSO is used as a good solution to achieve divergence. Also a mutation is applied on the best particle every iteration to reduce the risk of falling in a local optima.



**Figure:-5.6** Results of fragmented Two Layer Particle Swarm Optimization (FTLPSO).

**5.4.2 Performance Analysis on Various Datasets**

**Table:-2** Comparison Table of “COLUMN SCORE” on different Data Sets.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | FTLPSO | ACO - M | ACO - S | ACO – C |
| Data Set - 1 | 397.919 | 585.770 | 600.500 | **619.970** |
| Data Set - 2 | 415.607 | 588.940 | 598.020 | **602.620** |
| Data Set - 3 | 398.589 | 610.030 | 612.270 | **617.030** |
| Data Set - 4 | 457.297 | 589.400 | 600.940 | **607.480** |
| Data Set - 5 | 415.219 | 601.470 | 605.670 | **615.090** |
| Data Set - 6 | 422.507 | 601.350 | 606.840 | **613.620** |

**Table:-3** Comparison Table of “SUM OF PAIRS” score on different Data Sets.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | FTLPSO | ACO – M | ACO - S | ACO – C |
| Data Set - 1 | 30047 | 44652 | 46020 | **47510** |
| Data Set - 2 | 31861 | 45138 | 45950 | **46284** |
| Data Set - 3 | 29942 | 46198 | 46300 | **46868** |
| Data Set - 4 | 37055 | 45268 | 46368 | **46526** |
| Data Set - 5 | 32078 | 45998 | 46292 | **47000** |
| Data Set - 6 | 33836 | 45912 | 46386 | **47006** |

**Table:-4** Comparison Table of Length of “CONSENSUS” on different Data Sets.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| No. of dataset | LCS | FTLPSO | ACO - M | ACO - S | ACO - C | CLCS |
| Data Set - 1 | | | 20 | 18 | 24 | 25 | **27** | **57** |
| Data Set - 2 | | 20 | 24 | **36** | 35 | 35 | **53** |
| Data Set - 3 | | 20 | 18 | 42 | 42 | **43** | **61** |
| Data Set - 4 | | 16 | **48** | 35 | 35 | 34 | **55** |
| Data Set - 5 | | 19 | 33 | 35 | 35 | **36** | **60** |
| Data Set - 6 | | 13 | 33 | **38** | 36 | 37 | **54** |

**5.4 Comparison Analysis**

This presents the comparison of the table is based on the achieved column score, sum of pair and length of consensus. Results of ACO-C are compared with five Algorithms ACO-M, ACO-S, LCS, CLCS and existing algorithm FTLPSO. To overcame other tools and could keep the score high on same dataset. It also achieved the best Column Score, Sum of Pairs Score, Length of Consensus Obtained and Derived Consensus Sequence. After comparing the proposed method with other tools, it is proved that the results of ACO-C are good as compared to others.

**Table:-1** Resultant Table of proposed and Existing Approach on same Dataset.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| S.N. | ACO-C | FTLPSO | ACO-S | ACO-M |
| Achieved Column score (CS) | **619.970** | 397.919 | 600.500 | 585.770 |
| Achieved sum of pair (SOP) | **47510** | 30047 | 46020 | 44652 |
| Length of Conesus obtained (LOC) | **27** | 18 | 25 | 24 |

**Figure:-1** Comparison Table at level of CP, SOP and LOC.

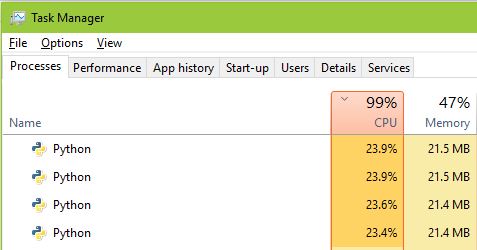
**Figure:-5.7** comparison graph of proposed and existing algorithm on the bases of column score(CS).

**Figure:-5.8** comparison graph of proposed and existing algorithm on the bases of Sum of Pair(SOP).

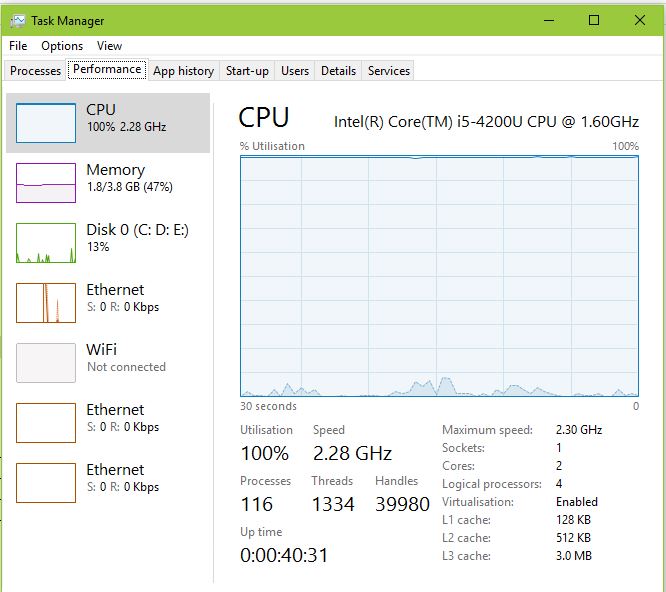
**Figure:-5.9** comparison graph of proposed and existing algorithm on the bases of consensus sequence

.

**Figure:-5.10** Performance graph of proposed algorithm on same dataset.



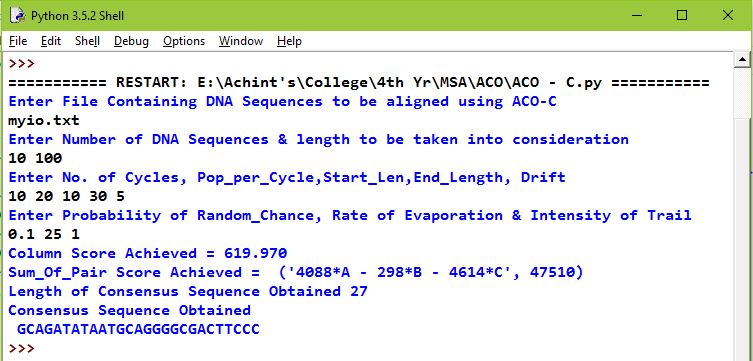
**Figure:-5.11** CPU and Memory uses by python.



**Figure:-5.12** System Performance after execution.

**5.5 Result OF Proposed Algorithm**

Ant colony Optimization with Cyclic Least Common Subsequence (ACO-C). The main idea of the system is that ants take a subsequence and move in an interval associated with each sequence, strengthening a pheromone trail when a close match (description of close match is the criteria, which can result in different variants of ACO) is found to a sequence at that position in other sequences. As the algorithm runs larger fragments of sequence are picked up by the ants.

****

**Figure:-5.13** Result of Ant colony Optimization with Cyclic Least Common Subsequence (ACO-C)

CHAPTER -6

**CONCLUSION AND FUTURE WORK**

* 1. **CONCLUSION**

This paper presents a DNA Sequence Alignment using Ant Colony Optimization with Circular Longest Common Subsequence (ACO-C) method to overcome the drawbacks of particle swarm optimization (PSO) and k-tuple fragmentation Algorithm. To improve its performance in solving multiple sequence alignment (MSA) problems. The standard PSO suffers from the trapping in local optima, and its disability to do better alignment for longer sequences. This Proposed Algorithm is best to work on long DNA Sequences. Results of ACO-C are compared with five Algorithms ACO-M, ACO-S, LCS, CLCS and existing algorithm FTLPSO. To overcame other tools and could keep the score high on same dataset. It also achieved the best Column Score (CS), Sum of Pairs (SOP) Score, Length of Consensus Obtained and Derived Consensus Sequence. After comparing the proposed method with other tools, it is proved that the results of ACO-C are good as compared to others.

**6.2 Future work**

Sequence alignment algorithms having a future driven scenario requirement, where the different algorithm take participate and outperform Sequence alignment over the multiple node participants. As a future path of this work is to focus on studying the efficiency of ACO Algorithm, decreasing the memory usage, CPU usage, and increasing the processing time. Proposed algorithm is working is serial manner, to make it in parallel manner is also going to be good future of proposed algorithm. These are the main interest as future works.