湖南大学

硕士学位论文

基于模式匹配的DNA多序列比对及相似性分析

姓名：王樱

申请学位级别：硕士

专业：计算机技术

指导教师：骆嘉伟；姜文君

20110924

摘 要

随着人类基因组计划（HumanGenomeProjeot，HGP）的顺利实施和信息技术的迅速发展，大量分子序列数据被人们发掘出来。对这些分子序列数据进行科学有效的分析和处理，让它们为人类疾病的诊断和治疗、疫情的预防、新药的开发等领域发挥更大的作用，已经成为人们愈加重视的研究话题，也是生物信息学的重要研究方向。生物信息学是多门学科相融合的新型的交叉学科。在生物信息学中，如何对基因序列进行有效且快速的比对，基因序列的相似性分析和进化关系分析都是其热门课题之一。

本文的主要工作是提出一种新的多序列比对算法——基于模式匹配的 DNA 多序列比对算法，并在其基础上进行基因序列的相似性分析。具体工作概括如下：

多序列比对是生物信息学中的一个基本问题。本文在模式匹配和 Aho-Corasick 搜索算法的理论基础上，深入分析研究了基于关键字树的 DNA 多序列比对算法，提出了一种新的多序列比对算法——基于模式匹配的 DNA 多序列比对算法。对该算法通过三组实验进行分析，并与原星比对算法、基于关键字树的 DNA 多序列比对算法进行比较。当序列相似度相对较低时，虽然该算法所用时间略长于基于关键字树的 DNA 多序列比对算法，但比对结果要优于基于关键字树的 DNA 多序列比对算法。当相似度很高的序列进行比对时，其比对的时间复杂度也优于另两种方法。实验结果表明了该算法的有效性。

序列相似性分析也是生物信息学中的基本问题之一，其分析结果可广泛应用于物种分类、结构和功能预测、物种进化分析等领域。本文将模式匹配方法应用于序列相似性分析，使用基于模式匹配的多序列比对结果，采用 Kimura 双参数模型和 Neighbor-joining 方法构建进化树。实验结果表明该方法得到了与事实相近的结果。

关键词：生物信息学；序列比对；星比对；模式匹配；相似性分析

Abstract

With the smooth execution for the Human Genome Project (HGP) and the rapid development of the information technology, a vast amount of molecular sequence data has come into existence. To make an effective scientific analyzing and processing of these alignment data so that they can play a great role in the diagnosis and treatment of human diseases, the prevention of tremendous epidemic diseases, and the development of new medicines has become a hot topic of conversation in common people's researches, and it is also an important research subject in the bioinformatics. The bioinformatics is a new interdisciplinary science of combined displines. How to make an effective and fast alignment of gene sequence and conduct similarity analysis and evolution relationship analysis on the basis of it is one of the hot topics in bioinformatics.

The major tasks of this thesis are to put forward a new multiple alignment sequence algorithm——a DNA way based on pattern matching and to conduct similarity analysis of gene sequence on the basis of this algorithm. The specific tasks are briefly summarized as follows:

The multiple sequence alignment is a basic issue in bioinformatics. On the theoretical basis of pattern matching and Aho-Corasick searching algorithm, this thesis makes a deep analysis and study of DNA multiple sequence alignment algorithm which is based on keyword tree, puts forward a new multiple sequence alignment algorithm——a DNA way based on pattern matching. This algorithm has been analyzed in experiments and compared with center star alighment algorithm and DNA multiple sequence alignment algorithm which is based on keyword tree. When the level of sequcence similarity is relatively low, the alighment result is superior to the DNA multiple sequence alignment algorithm based on keyword tree with the weakness of the time occupied is a little more. When the sequnces with high level of similarites are aligned, the alignment time complexity is also superor to the other two methods. The results of experiments testified the effectivity of this algorithm.

The analysis of sequence similarity is also one of the basic issues in bioinformatics. The result of analysis can be widely used in species classification, the prediction of structures and functions and species evolution analysis. This thesis puts the pattern matching method into the sequence similarity analysis, and uses the sequence alignment algorithm which is based on pattern matching to make sequence alighment. The results of alignment are used to construct the evolution tree in Kimura double parameter pattern and Neighbor-joining way. The results of experiments testified that the algorithm had got the result which is near the facts.

Key Words: Bioinformatics; sequence alignment; center star method; pattern matching; similarity analysis

插图索引

图 2.1 DNA 分子的双螺旋结构 8
图 2.2 遗传信息的传递和表达 9
图 2.3 分子生物学的中心法则 10
图 2.4 矩阵元素 M[i,j]的来源 16
图 2.5 包含 4 个物种的有根树 19
图 2.6 包含 4 个物种的无根树 20
图 4.1 6 种西藏根瘤菌的进化树 40
图 4.2 11 种物种的进化树 41
图 4.3 8 种 H5N1 型禽流感病毒的 HA 片断基因的进化树 42

附表索引

表 2.1 蛋白质的氨基酸名称及字母符号 7
表 3.1 6 种西藏根瘤菌 RDNA 全序列 28
表 3.2 6 种西藏根瘤菌 RDNA 全序列比对运行时间统计（单位：ms） 28
表 3.3 6 种西藏根瘤菌 RDNA 全序列碱基对匹配情况对比 29
表 3.4 6 种西藏根瘤菌 RDNA 全序列比对结果情况对比 29
表 3.5 11 个物种的 β-球蛋白基因第一个外显子 29
表 3.6 11 个物种的 β-球蛋白基因第一个外显子比对运行时间统计（单位：ms） 30
表 3.7 11 个物种的 β-球蛋白基因第一个外显子碱基对匹配情况对比 30
表 3.8 11 个物种的 β-球蛋白基因第一个外显子的比对结果情况对比 31
表 3.9 8 种 H5N1 型禽流感病毒 HA 片段 31
表 3.10 8 种 H5N1 型禽流感病毒 HA 片段序列的比对运行时间统计（单位：ms） 31
表 3.11 8 种 H5N1 型禽流感病毒 HA 片段序列的碱基对匹配情况对比 32
表 3.12 8 种 H5N1 型禽流感病毒 HA 片段序列的比对结果情况对比 32
表 4.1 Kimura 双参数模型 37
表 4.2 4 种西藏根瘤菌 RDNA 全序列的进化矩阵 40
表 4.3 11 个物种第一个外显子的进化矩阵 41
表 4.4 8 种 H5N1 型禽流感病毒的 HA 片断基因的进化矩阵 42

第1章 绪 论

1.1 研究背景与意义

随着人类基因组计划的顺利实施和信息技术的迅速发展，GeneBank、EMBL 和 DDBJ 国际三大核酸序列数据库数据量呈指数增长。生物学家、数学家和计算机科学家都面临着一个相同的并且严峻的问题，如何利用、表达这些数据进而分析与解释基因序列间的潜在关系，从中发掘出对人类有利的信息。为了迎接这一挑战，一门涉及生物、数学、物理、化学、计算机科学等诸多科学的新型学科应运而生，并且日益成为二十一世纪自然科学的核心领域之一。

生物信息学的主要研究对象是 DNA 序列和蛋白质序列，主要通过分类、分析和检索核苷酸序列或氨基酸序列，获取基因编码和调控、代谢途径、DNA 和蛋白质结构功能及其相互关系等各方面的知识。所以在生命起源、生物进化以及细胞、器官和个体的出现、生长、病变、消亡等生命科学问题中，生物信息学都起着非常重要的作用。生物信息学是发现生命科学问题中的基本规律和时空联系，发掘生物序列数据中蕴含的生物学意义的交叉学科[1]。

在生物信息学中，序列比对是最基本、最重要的操作。对于基因序列，通过比对可以推测出哪个基因家族可能包含该序列，并可以推测出该序列可能具有的生物学功能；对于蛋白质序列，通过比对可以推测出该序列可能的功能和结构，并可以找出与它同源的蛋白质序列。所以在生物信息学中，序列比对具有非常重要的意义和实用价值。目前，国际上提出了众多经典的比对算法，也开发了众多的序列比对软件。但对于同一组序列，不同的软件采用不同的序列比对算法，其运算速度和比对结果都有很大差异。有些软件考虑了比对结果而运行时间较长，而有些软件运算速度很快比对结果却不理想。一般情况下两者不能同时兼得。所以，对于序列比对算法的研究还有待继续深入。

序列比对的主要任务是采用某种算法比较 DNA（蛋白质）序列，发掘序列之间的相似性和相异性。在生物信息学中，DNA 序列或蛋白质序列的相似性主要表现在序列、结构和功能三个方面。通常情况下，序列决定结构，结构决定功能。所以，序列相似性研究主要表现在两个方面，一方面通过序列相似性分析发掘序列的结构和功能；另一方面通过序列相似性分析发现序列间的进化关系。许多有科学价值的研究都依赖于利用计算机进行序列的相似性分析。

1.2 国内外研究现状及发展趋势

1.2.1 多序列比对

多序列比对是双序列比对的一般性推广。由于核酸数据库容量的增长呈指数级，序列比对的数量通常都会远远大于两个，使用动态规划算法来解决比对问题已经是不可行的了，这使得多序列比对成为一 NP 难题。为了解决这一难题，人们提出了许多近似算法。

1．动态规划算法

多序列比对最早采用的是动态规划算法来解决。动态规划算法中最为经典的是 Needleman-Wunsch 算法，其解决思路是把整个问题分解成多个相互联系的子问题，通过依次解决每个子问题，从而解决整个问题。动态规划算法最初用于求解两个序列的比对，当把动态规划的基本思想推广到多序列比对时，3 个很短序列的比对还可以顺利进行。比对序列的数量如果超过 3 个，由于需要很大的存储空间和很长的运行时间，比对根本无法进行下去。所以多序列比对问题不能采用动态规划算法来解决。Carrillo 和 Lipman 等人对该算法进行了改进，提出了 Carrillo-Lipman 算法，通过减少存储空间将序列比对的数量提高到 10[2]。2004 年，唐玉荣等人对动态规划算法进行了优化[3]，与基本动态规划法敏感性相同，但降低了算法的时间复杂度，并在减少存储空间方面也有一定的效果。

2．启发式算法

目前，绝大多数算法属于启发式算法，包括星比对算法，渐进式比对算法，迭代细化方法等。其中应用最早的是星比对算法，而应用最广并且效果较好的是渐进式比对算法。Hogeweg 和 Hesper 首先提出渐进式比对算法[4]，而后 Feng 和 Taylor 对其加以完善[5.6]。与动态规划算法相比，该算法在计算速度、存储空间和序列数目等方面都更加优良。并且，渐进式比对算法能够直接用于构造进化树，反映序列间的进化关系。2005 年，段敏等人提出了一种用减少序列比对过程中总评分的方法来达到局部优化目的的多序列比对算法[7]。启发式算法虽然在一定层度上减少了算法的运行时间和存储空间，但都有一些不足之处。星比对算法中，无论采用何种方法并不能保证找到的序列是最好的中心序列。渐进式比对算法中，构造的指导树有时不一定真正反映系统的进化信息，根据指导树渐进比对容易产生局部最优化问题。迭代细化算法中，无法采用何种迭代策略得到的结果最优。

3．随机算法

多序列比对中，应用最多的随机算法有遗传算法、模拟退火算法和粒子群算法等。

遗传算法是一种全局意义上的自适应随机搜索方法，它借鉴生物进化规律，模拟生物进化过程中的一系列事件，包括突变、交配和选择，最终得到一个优化解。模拟退火算法则是模拟物理中的退火过程并结合复杂系统中的组合优化之间的相似性来寻找最优解。2008 年，向昌盛等人提出了将遗传算法和模拟退火算法相结合的遗传退火进化思想[8]，设计了运用该思想进行多序列比对的算法过程，实验结果表明该算法是行之有效的。2011 年，徐小俊等人针对粒子群优化易陷入局部最优、收敛速度慢的现象，提出了一种分段取值惯性权重（SW）方法[9]，该方法在解决多序列比对问题时可以有效地避免算法早熟，并提高解的精度。

4．分治算法

分治算法是把一个大问题分解成若干个小问题来解决。Stoye 提出了一种新的分治算法 DCA[10]，将 Carrillo-Lipman 算法引入进来。在不影响特征表现的前提下，把序列分割成完全满足 Carrillo-Lipman 算法长度要求的子序列，使用 Carrillo-Lipman 算法进行序列比对。2000 年 Stoye 又提出了一种 OMA 算法[11]，以达到减少存储空间的目的。2009 年，业宁等人设计了一个 DCA-ClustalW 算法来解决多序列比对问题[12]，从纵向和横向两个方面将复杂问题简单化，并在 BaliBase 基准数据集上测试了算法的可行性。

5．其他算法

2006 年，陈娟等人给出了多重序列比对的蚁群算法[13]，结果显示蚁群算法可以有效解决多重序列比对问题并具有自适应性、鲁棒性等特点。而文献[14,15]针对蚁群算法易于陷入局部最优解、收敛速度慢等问题，提出了改进的方法。

1.2.2 序列相似性分析

生物信息学中的另一个研究热点就是序列相似性分析。起初，序列相似性分析基本采用序列比对的方法来实现。最早用于相似性分析的序列比对算法是 Gibbs 的点阵图法[16]。其后，Needleman 和 Wunsch 提出了进行全局比对的 Needleman-Wunsch 算法[17]，Smith 和 Waterman 提出了进行局部比对的 Smith-Waterman 算法[18]。目前的许多算法都是在这些算法上的改进。由于序列比对算法复杂、计算量大，2000 年 Randic 等人首次提出非序列比对的方法实现相似性分析，利用矩阵将复杂问题简单化，将序列比对转化为矩阵不变量的比较[19]。文献[20,21,22]都在传统的矩阵不变量上进行改进，减少其计算量，但又使其值接近于传统的矩阵不变量。另外一些学者在序列相似性分析中引入几何图形表示，并从中提取不变量。汪挺松引入了图形曲率，作为生物相似性比较的不变量[23]，计算量也大大降低。李梅等人采用基于 DTW 的 DNA 序列相似性度量方法能够有效地解决动态规划算法对空位罚分的主观性依赖[24]。唐晓婵采用 4D 图形的几何中心表

示 DNA 序列比较的不变量[25]，在进行序列相似性分析时能够得到较好的效果。还有一些学者将信息理论方法作为相似性分析的基础。刘芳提出了修正的广义信息距离，即一种新的基于信息离散度的序列差异度量方法[26]。该方法既适用于相似程度很高的序列，也适用于差异性较大的序列。

1.2.3 进化树构建

构建进化树就是通过进行序列相似性分析，将物种合理地分成不同的群体，同一群体中的物种相似性高于不同的群体，由此来建立物种的进化关系。构建进化树的方法很多，主要分为两类：距离矩阵法和非距离矩阵法。距离矩阵法有 UPGMA 算法[27]、Fitch-Margoliash 算法[28]和 Neighbor-joining 算法[29]等。非距离矩阵法有最大简约法[30]和最大似然法[31]等。Satoshiota 等人提出了采用“divide-and-conquer”机制的 NJML 方法[32]，该方法将最大似然法引入 Neighbor-joining 算法中，从而找到最佳的拓扑图形。Vincent Ranwez 等人提出了 TripleML 法[33]，通过提高序列间距离值的精确度来提高构建 Neighbor-joining 进化树的精确度。为了提高 Neighbor-joining 算法的效率，谭严芳等人在使用 Neighbor-joining 算法构建进化树时引入了校正距离和 Kimura 两参数模型[34]。针对使用 UPGMA 算法构建进化树的不唯一问题，徐立业等人提出的不加权算术平均组群方法解决了这一问题[35]。另外，也有人将几何图形表示应用到进化树的构建中。这种方法不需要进行序列比对，不需要考虑物种在进化过程中的突变和颠覆等，因此更适用于基因组数据的相似性分析。郑文新通过 z 曲线计算获得几何中心和协方差矩阵，将最大特征值 λ 和相应向量之间的夹角余弦值表示成序列之间的距离，使用 PHYLIP 软件包中的 UPGMA 算法构建进化树[36]。2006 年廖波等人将模糊聚类的传递闭包法运用于 2D、3D 图形中来构建进化树[37]。张建辉使用序列的 3D 图形计算距离矩阵，将得到的矩阵使用 PHYLIP 软件包中的 Neighbor-joining 算法构建进化树[38]。张惜珍等人将 N 曲线和可凝聚的层次聚类算法相结合来构建进化树[22]，构建的进化树与使用 PHYLIP 软件包的构建结果一致。柳菁筠等人引入了图论的概念，使用一个带权值的连通图来表示距离矩阵，再使用 Kruskal 算法构建进化树[39]。苏志忠将分子序列的频率向量作为样本向量，提出了基于信息相异性的模糊聚类算法构建系统树[40]。李刚成等人提出的基于模糊聚类的构树方法，引入了 DNA 序列的 4D 表示方法[41]，降低了聚类过程中的计算复杂度。

1.3 本文所做的工作

本文以生物信息学为背景，重点研究 DNA 多序列比对方法、基于序列比对的相似性分析方法。本文的主要工作有：

1. 对当前国际上流行的多序列比对算法进行了细致分析和深入研究，在模式匹配理论和 Aho-Corasick 搜索算法的基础上，结合基于关键字树的 DNA 多序列星比对算法[42]，提出了基于模式匹配的 DNA 多序列全局比对方法，并实现了这个方法。最后通过实验和原始的星比对算法、基于关键字树的 DNA 多序列星比对算法进行了比较，验证了本方法的有效性和可行性。

2. 将基于模式匹配的 DNA 多序列比对方法应用于相似性分析，使用该方法得到的比对结果，采用 Kimura 双参数模型和 Neighbor-joining 方法实现相似性分析。同样通过三组实验证明了该方法的有效性。

1.4 论文结构

论文分五章，具体安排如下：

第一章主要介绍序列比对及相似性分析的研究背景和意义，以及多序列比对、相似性分析、进化树的构建在国内外的研究现状，最后概括地介绍了本人所做的工作。

第二章主要介绍生物信息学的基本概念，序列比对中涉及到的一些基本概念和问题，并介绍了一些经典的序列比对算法。最后介绍了分子进化的基本概念和怎样构建进化树。

第三章在模式匹配理论和 Aho-Corasick 搜索算法的基础上，针对基于关键字树的 DNA 多序列星比对算法，提出了基于模式匹配的 DNA 多序列比对算法，并通过实验和原始的星比对算法、基于关键字树的 DNA 多序列星比对算法进行了比较分析。

第四章将基于模式匹配的 DNA 多序列比对方法应用于相似性分析，同样通过三组实验证明了该方法的有效性。

最后，对本人所作的工作进行了总结，并分析了所提出的方法的不足之处，以及改进方向。

第 2 章 生物信息学相关知识

2.1 生物信息学基本概念

2.1.1 生物信息学的主要研究内容

生物信息学的主要任务是分析、处理和研究 DNA 序列（蛋白质序列）数据中所包含的各种生物学信息。生物信息学的研究内容主要包括序列比对、蛋白质结构比对和预测、基因识别、分子进化和比较基因组学、序列重叠群装配、基于结构的药物设计等[43]。

1. 序列比对

序列比对是生物信息学的最基本的一个问题，主要是对两个或两个以上的 DNA 序列（蛋白质序列）进行相似性的比较。通过序列比对可以从相互重叠的序列片段中重构 DNA 的完整序列，在数据库中搜索相关序列和子序列，预测未知序列的结构和功能等。

2. 蛋白质结构比对和预测

蛋白质结构比对是对多个蛋白质分子空间结构的相似性进行比较。一般认为具有相似功能的蛋白质结构一般相似。通过对蛋白质结构的比对，预测未知蛋白质的结构和功能。

3. 基因识别

基因识别是正确识别给定基因在基因组序列中的准确位置。对于真核生物基因组序列，编码区仅占极少的部分，要正确识别出其精确位置还非常困难，仍有大量的工作要做。

4. 分子进化和比较基因组学

分子进化是分析同一基因序列在不同物种中的相似性和相异性，研究物种的进化关系，构建物种的进化树。一般使用 DNA 序列或蛋白质序列来完成，也可通过相关蛋白质的结构比对来研究分子进化。其研究基础是假定相似种族在基因上具有相似性。通过比较可以在基因组层面上发现哪些是不同种族中相同的特性，哪些是不同的。

5. 序列重叠群装配

序列重叠群装配是逐步把较短的序列的相同部分重叠起来构成一个重叠群，这样会得到一个逐步变长的序列，直至得到完整序列这个过程才会停止。已经证明，这是一个 NP 难问题。

6. 基于结构的药物设计

基于结构的药物设计是分析生物序列与人类疾病之间的关系，然后研发相应的药物

对疾病进行预防和治疗。目前，基于蛋白质结构的药物设计在生物信息学研究中发挥着重要的作用。

7. 其它

生物信息学的研究内容除了上述之外，还有其他一些重要领域，如基因芯片设计、基因表达谱分析等等。

2.1.2 核酸和蛋白质

核酸是一种一维高分子链，链中包含有 4 种单体，每个单体叫做核苷酸。核酸中携带着遗传信息，遗传信息主要表现在核苷酸的排列次序上。根据核苷酸类型的不同，核酸分为脱氧核糖核酸（DNA）和核糖核酸（RNA）。核苷酸由磷酸、脱氧核糖或核糖和碱基组成。构成核苷酸的碱基分为嘌呤和嘧啶两大类。前者主要指腺嘌呤（adenine，A）和鸟嘌呤(guanine,G),DNA 和 RNA 中均含有这两种碱基。后者主要指胞嘧啶(cytosine，C)、胸腺嘧啶（thymine，T）和尿嘧啶（uracil，U），胞嘧啶存在于 DNA 和 RNA 中，胸腺嘧啶只存在于 DNA 中，尿嘧啶则只存在于 RNA 中。其中，DNA 是储存、复制和传递遗传信息的主要物质基础，RNA 在蛋白质合成过程中起着重要作用[44]。

蛋白质是构成生物体的直接元素，使用不同的蛋白质构造出了不同的生物体。蛋白质由二十种氨基酸通过肽键连接而成。这二十种氨基酸是蛋白质的基本单位，赋予蛋白质特定的分子结构形态[44]。表 2.1 列出了这二十氨基酸的名称、字母表示和符号。

表 2.1 蛋白质的氨基酸名称及字母符号

名称	三字母	符号
丙氨酸（alanine）	Ala	A
精氨酸（arginine）	Arg	R
天冬酰胺酸（asparagine）	Asn	N
天冬氨酸（aspartic acid）	Asp	D
半胱氨酸（cysteine）	Cys	C
谷氨酰胺酸（glutanine）	Gln	Q
谷氨酸（glutamic acid）	Glu	E
甘氨酸（Glicine）	Gly	G
组氨酸（histidine）	His	H
异亮氨酸（isoleucine）	Ile	I

续表 2.1

名称	三字母	符号
亮氨酸（leucine）	Leu	L
赖氨酸（lysine）	Lys	K
甲硫氨酸（methionine）	Met	M
苯丙氨酸（phenylalanine）	Phe	F
脯氨酸（proline）	Pro	P
丝氨酸（serine）	Ser	S
苏氨酸（threonine）	Thr	T
色氨酸（tryptophan）	Trp	W
酪氨酸（tyrosine）	Tyr	Y
缬氨酸（valine）	Val	V

2.1.3 遗传信息传递与表达

在生物进化过程中，主要是通过 DNA 实现遗传信息的传递和表达。DNA 是一种由多种不同的脱氧核苷酸组成的高分子有机化合物，是一种链式结构。其中，脱氧核苷酸又是由磷酸、脱氧核糖和含氮碱基三部分组成。图 2.1 给出了 DNA 分子的双螺旋结构。脱氧核糖与磷酸形成的两条主链构成了 DNA 分子的骨架，两条主链内侧碱基对一一对应起来，在碱基对之间由氢键进行连接。碱基配对遵循 A 与 T 配对、C 与 G 配对的原则。

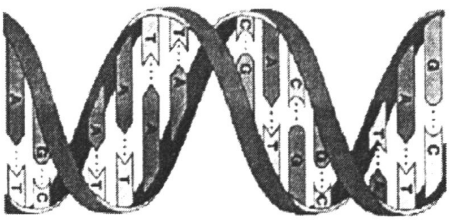

图 2.1 DNA 分子的双螺旋结构

碱基间配对的特异性是 DNA 精确复制的基础。在 DNA 分子复制过程中，首先使用解螺旋酶将两条主链分离，再将其中的一条主链作为模板，通过聚合酶按照碱基配对的原则生成一条新的主链，重新构成双螺旋结构，这样 DNA 分子复制就完成了。由于遗传信息主要表现在核苷酸的排列顺序上，当被复制的主链确定后，该主链上的核苷酸序列顺序也就确定，基于碱基配对原则，复制生成的核苷酸序列顺序也被自动确定下来，从而 DNA 分子的遗传信息也被进行了正确地复制。因此 DNA 分子的复制揭示了遗传

信息的传递。

蛋白质并不能直接获取 DNA 分子中的遗传信息。遗传信息通过 DNA 分子传递给蛋白质主要是通过两个步骤进行的。第一步是在通过以 DNA 为模板合成 RNA 的过程中，遗传信息传递给了 RNA，这个过程称为转录，合成的 RNA 称为信使 RNA（mRNA）。第二步在通过以 RNA 为模板合成蛋白质的过程中，遗传信息传递给了蛋白质序列，这个过程称为翻译。在翻译过程中，信使 RNA 的每三个相邻的碱基称为密码子，会转变成一种氨基酸。

遗传信息的传递和表达如图 2.2 所示。

图 2.2 遗传信息的传递和表达

2.1.4 变异

变异是指在生物进化过程中 DNA 序列的某些碱基发生了改变。变异可以分为三类：

替代：在生物进化过程中生物序列中的某一碱基被其他碱基所替代。

插入或删除：在生物进化过程中增加或者删除一个或多个碱基。

重排：DNA 序列或蛋白质序列的一些片段在合成过程中发生了连接顺序的改变。

在实际研究过程中，变异起着非常重要的作用。变异不仅会造成遗传的变异和疾病，另外物种的多样化也是变异造成的。

2.1.5 生物信息学的中心法则

分子生物学的中心法则首先由 Francis Crick 提出。它系统的概括了 DNA 和蛋白质这两大生物大分子之间是如何进行信息传递的。总的来说，DNA 携带着生物进化的遗传信息，在一定条件下 DNA 可以非常精确地进行自我复制。为此首先要把遗传信息“转录”到信使 RNA 上。然后把信使 RNA 上的遗传信息翻译到蛋白质上。新生的蛋白质折叠形成特定的三维形状，在生命过程中发挥其特有的功能。因此，从 DNA 到蛋白质的信息传递过程实际上是一个单向信息流动过程，这个过程同 DNA 本身的信息传递一起构成了分子生物学的中心法则[45]。分子生物学的中心法则如图 2.3 所示。后来，人们发现并不是所有的生物都遵循这一法则。有些病毒的遗传信息是存储在 RNA 中，遗传信息的传递是从 RNA 反转录到 DNA 上，这样才能进行 DNA 的自我复制。

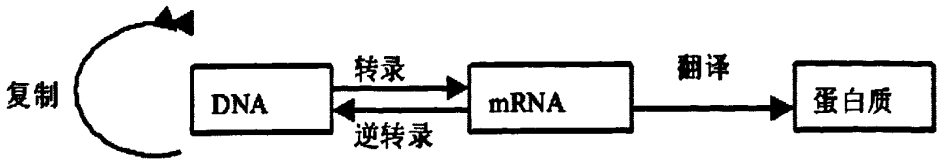

图 2.3 分子生物学的中心法则

2.2 序列比对

2.2.1 序列比对概述

科学研究中，比较是最常见的方法之一。为了寻找对象间的相同处和差异处或者发掘对象可能具备的特性，通常会采用比较的方法来完成。在生物信息学研究中，比较就是将多个相似的序列进行比对。序列比对源于进化学说。如果两个序列非常相似，那么可以推测这两个序列可能具有相同的祖先，由其祖先经过不同的基因替代、增加、删除和重排等变异过程而演变而来。另外，通过序列比对还可以推测给定蛋白质序列可能具有的结构和功能。所以，序列比对可应用于二级结构预测、蛋白质的功能域识别、基因识别等方面的研究。

进行序列比对就是采用某种特定的数学模型或算法，找出序列之间的最大匹配碱基数[46]，即在两个或多个字符串序列中插入空位“-”以达到匹配的字符数量最大。例如，对两个序列 TCGAAGCTGGT 和 TCGAAGCGGT 进行序列比对。

空位“-” 插入前两序列匹配如下：

```
T C G A A G C T G G T
| | | | | | | | | |
T C G A A G C G G T
```

其中有 8 个相同碱基。

空位“-” 插入后两序列匹配如下：

```
T C G A A G C T G G T
| | | | | | | | | | |
T C G A A G C - G G T
```

当两序列采用插入空位的方法进行比对后有 10 个相同碱基，相对于采用不插入空位的方法而言增加了匹配的数目，从这里可以看出插入空位是非常有必要的，其过程也形象的反映了生物进化的过程。

序列比对的实现一般都依赖于某个数学模型。不同的数学模型可能反映序列结构、

功能、进化关系等不同的特性。很难断定某个数学模型的好坏，也很难断定某个数学模型的对错，它只从某个角度反映序列的生物学特性。

2.2.1.1 多序列比对问题

综上所述，我们可以用一个五元组来描述多序列比对问题，见式(2.1)：

$$MSA = (\Sigma, S, A, O, F) \tag{2.1}$$

其中：1) Σ表示多序列比对的符号集，其值为$\Sigma 1 \cup \{-\}$；$\Sigma 1$表示有限符号集，蛋白质序列比对时$\Sigma 1 = \{A,C,D,E,F,G,H,I,K,L,M,N,P,Q,R,S,T,V,W,Y\}$，DNA 序列比对时$\Sigma 1 = \{A,T,C,G\}$，—表示空位符，即比对过程中需要插入的空位。

2) S表示待比对序列集，蛋白质序列比对时每条序列由氨基酸组成，DNA 序列比对时，每条序列由碱基组成，序列长度不等。$S = \{S_i \mid i = 1,2,...,m\}$，$S_i = (c_{ij} \mid j = 1,2,...,l_i)$，其中，m 为序列个数，$c_{ij}$为序列$s_i$中的第 j 个碱基，$l_i$为第 i 个序列的长度。

3) A表示结果矩阵，$A = (a_{ij})_{m \times n}$，$a_{ij} \in \Sigma$。在结果矩阵中，第 i 行表示第 i 条序列，矩阵的每 j 列表示第 j 个碱基的比对结果。序列中的碱基顺序不能在比对前后发生变化。

4) O表示比对操作集，$O = \{insert\_gap, delete\_gap\}$，即插入空位和删除空位操作。

5) F为比对算法，确定空位插入和删除操作的特定位置。

2.2.1.2 空位罚分

在序列比对的过程中，为了使序列比对的结果更符合某种期望值，通过引入空位来弥补序列的插入或删除操作。但是，不能无限制的引入空位，否则会使比对结果缺乏生物学意义。为了限制空位的插入，通常的方法是在插入空格时对总分值进行扣分。扣除的分值为罚分值，这个用罚分的方法来限制空位的插入称为空位罚分。因此，序列比对的结果得分是两个序列之间匹配残基的总分值与空位罚分的总和[46]。

假设，使用 s_1 和 s_2 表示待比对的序列，使用 s_{10} 和 s_{20} 表示比对之后的结果，使用 L 表示序列比对后的长度，一般有三种空位罚分规则：

1. 空位罚分

最简单的一种罚分规则就是空位罚分，当序列插入一个空位就给定一个固定的罚分值 Wg，对于整个序列而言，总的空位罚分等于插入的空位数 Rg 乘以每个空位的罚分

值 Wg，即 Wg×Rg[47]。

空位罚分不会增加额外的运行时间，所以是最简单的罚分规则。但是，基因序列中碱基的突变频率在实际生物进化过程中是根据位点的不同而不同的，而空位罚分中每个位点的罚分值都相同，与其实际的生物学意义有所出入。

2. 恒定空位罚分

这种罚分规则针对的不是每一个空位，而是每一个空格，这里把相连的空位称作一个空格。根据序列中插入的空格进行罚分，每插入一个空格罚分值为 Wg[47]。具体操作如下：

1) 假设无论匹配或不匹配，其得分值由 σ 表示，则 $\forall x,\sigma(x,-)=\sigma(-,x)=0$；

2) 比对得分的计算公式为：

$$\sum_{i=1}^{L}\sigma(s_{10}[i],s_{20}[i])+W_g\times gaps \tag{2.2}$$

其中 gaps 表示空格的数目。

恒定空位罚分虽然避免了单独依赖空位长度进行罚分的缺陷，但当插入过多相连空位时这种罚分规则却无法进行限制，可能导致序列片断被相连空位分裂。这就需要一种罚分规则与空位长度是紧密联系的，其罚分值既不单纯依赖于空位长度也不忽略空位长度。

3. 仿射空位罚分

该罚分规则把空位罚分分成开放空位罚分和扩展空位罚分两个部分[47]。设定 q 为空位长度，Wg 为开放空位罚分值，Ws 为扩展空位罚分值，W 为总罚分值，则：$W=W_g+q\times W_S$，这样比对得分的计算公式为：

$$\sum_{i=1}^{L}\sigma(s_{10}[i],s_{20}[i])+W_g\times gaps+W_s\times spaces \tag{2.3}$$

其中 gaps 表示空位的数目，spaces 表示空格的数目。

在实际生物学研究中，多次插入和删除一个空位比相连的多个空位的插入和删除的几率要小，所以，放射空位罚分更具有生物学意义。

2.2.1.3 计分矩阵

在序列比对中，通常用一个矩阵来记录每种变异的分值，这个矩阵称为计分矩阵。选择不同的计分矩阵，序列比对的结果也会不同。最简单的计分矩阵是单一矩阵，也称为稀疏矩阵。采用该矩阵，只需要检测序列之间对应位点的碱基是否完全相同，相同碱

基的分值为 1，不相同则为 0。这种只考虑碱基同一性的矩阵具有很大的局限性。

为了能够更好的体现生物学特征，就需要设计更优的计分矩阵。PAM 是第一个被广泛使用的最优矩阵。一个 PAM 表示有 1%的氨基酸发生改变，即进化的变异单位。这样并不能说明经过 100 次的变异又回到了初始状态。因为不同的氨基酸的出现频率是不相同的，而氨基酸的取代是基于氨基酸的出现的。所以有些位点可能会发生多次取代，而有些位点可能没有任何变化。经过多次实验发现，在蛋白质中已经发现的替代大大倾向于不影响蛋白质功能的替代。Dayhoff 等人用非常相近的序列收集并推广了 PAM。虽然 Dayhoff 等人只发表了 PAM250，但同样可以外推到其他 PAM 值。一般情况下，相似度很高的序列使用较低值的 PAM 矩阵，相似度低的序列使用较高值的 PAM 矩阵。

除了 PAM 矩阵外，BLOSUM 矩阵也被广泛使用。BLOSUM 矩阵同样也使用编号来区别不同的 BLOSUM 矩阵，这里编号的作用主要是区别序列的相同程度。例如，BLOSUM62 矩阵一般使用在比对序列中至少有 62%的相同比例。所以，在使用 BLOSUM 矩阵时刚好与 PAM 矩阵相反。高值的 BLOSUM 矩阵一般使用在相似程度很高的序列中，低值的 BLOSUM 矩阵则使用在差异性较大的序列中。

2.2.1.4 序列比对结果的评判标准

评价多序列比对结果的好坏到目前为止还没有一个公认的标准[48]。很多情况下都是通过实验并计算两个分值 the sum of pair score（SPS）和 the column score（CS）[49]来评价多序列比对的结果。假设实验中待比对的序列个数用 N 表示，每条序列的长度用 M 表示，参考序列的长度用 Mr 表示，第 i 条序列中第 j 个碱基用 c_{ij} 表示，则这两个分值的计算公式如下：

1．SPS 分值

同一列上的一对碱基 c_{ij} 和 c_{ik}，定义 P_{ijk}，如果 c_{ij} 与 c_{ik} 相同，则 P_{ijk} 为 1，反之为 0。则每一列的值 S_i 的计算公式为：

$$S_i = \sum_{j=1, j\neq k}^{N} \sum_{k=1}^{N} P_{ijk} \tag{2.4}$$

假设 S_r 值为参考数据对应的 S_i 值。则 SPS 值计算公式为：

$$SPS = \sum_{j=1}^{M} S_i / \sum_{j=1}^{Mr} S_{ri} \tag{2.5}$$

如果没有标准比对库作参考，SPS 的计算公式为（这种方法一般适用于同一比对序列在多种方法下的比较）：

$$SPS = \sum_{i=1}^{M} S_i /(M \times N \times (N-1)/2) \tag{2.6}$$

2. CS 分值

如果每个序列同一列上的所有碱基都相等，则$c_i=1$，否则$c_i=0$，则 CS 分值的计算公式为：

$$CS=\sum_{i=1}^{M}c_i/M \tag{2.7}$$

由上述可知，CS 分值和 SPS 分值是使用不同的对象来进行评定。CS 分值是计算所有序列准确对齐的比率，是针对所有序列而言的。SPS 分值是计算碱基对准确对齐的比率，是针对比对序列中的两两比对而言的。本文采用 SPS 分值将本算法与其它算法进行比较。

2.2.2 序列比对算法

目前，很多序列比对算法都是以动态规划算法为基础，在运算速度和存储空间方面考虑不同程度的改进。序列比对算法有几种不同的分类方法。根据比对序列数目的多少，可以把序列比对分为双序列比对和多序列比对；根据序列的比对范围的大小，可以把序列比对分为全局序列比对和局部序列比对。

2.2.2.1 局部序列比对和全局序列比对

局部序列比对考虑序列的局部相似性，是寻找序列部分相似区域的方法。局部序列比对主要应用于蛋白质序列的比对，相对于全局比对更加灵敏，更具有生物学意义。全局序列比对比对的是序列的整体，是从全局的范围去考虑序列的相似性。全局序列比对主要应用于序列之间的同源关系，预测蛋白质的结构和功能等。

2.2.2.2 双序列比对

双序列比对就是找出两条 DNA 序列或蛋白质序列的最大相似性匹配，其寻找的过程是基于某种算法或模型的。多序列比对和序列数据库搜索都是基于双序列比对的。目前，最经典的双序列比对算法有点阵图法和动态规划算法。

1. 点阵图法

最简单的双序列比对算法就是点阵图法。该方法将待比对的序列放在一个二维平面上，一条序列横向地放在平面的上方，一条序列纵向地放在平面的左方。两个序列的任何两个相同碱基的交叉位置都进行标注一个点。最后将平行于对角线上的点连接起来就构成了两序列的比对结果[16]。

点阵图法能够很形象地显示出序列的插入和删除，并且两条序列之间所有匹配的碱基序列都可以用点阵图直观的反映出来。但是由于序列长度都是以千计数，用现有的点阵计算机程序显示真实的比对序列是不太现实的，所以更多的会使用其他比对方法来实现。

2. 动态规划算法

动态规划算法首先由 Needleman 和 Wunsch 提出并得到广泛的应用和改进，逐渐成为计算生物学中最重要的理论基础之一。最经典的动态规划算法是 Needleman-Wunsch 算法和 Smith-Waterman 算法。所有的全局序列比对算法都是基于 Needleman-Wunsch 算法的，而 Smith-Waterman 算法是在 Needleman-Wunsch 算法基础上作了改进，主要应用于局部序列比对。下面对动态规划算法作详细的介绍。

给定序列 s_1 和 s_2，长度分别为 m 和 n。$s_1[1...i]$和 $s_2[1...j](1\leq i\leq m,\ 1\leq j\leq n)$分别表示 s_1 和 s_2 的前缀子序列。s_1 和 s_2 的比对结果包含了 $s_1[1...i]$和 $s_2[1...j]$的比对结果，这是一个递归的关系。从这个关系可以看出求解其子序列的最优解是求解两序列全局比对的前提。通过这个递归关系从而得到整个序列的最优值。再根据获得最优值的路径回溯回去，得到序列的最优比对结果。

动态规划算法的基本步骤是使用一个二维矩阵存储两个序列的相似分值，然后根据这个矩阵中的分值进行回溯得到序列的最优比对。假设对序列 s_1 和 s_2 使用动态规划算法进行比对，其长度分别为 m 和 n。首先需要构建一个大小为(m+1)×(n+1) 二维矩阵，矩阵中的元素 $M[i,j](0\leq i\leq m,0\leq j\leq n)$表示其前缀子序列 $s_1[1...i]$和 $s_2[1...j]$的最优比对得分。矩阵的第 0 行和第 0 列都表示其前缀子序列 $s_1[1...i]$和 $s_2[1...j]$和空位“-”的比对分值。因此，矩阵中各元素的初始值为：

$$M[0,0]=0 \tag{2.8}$$

$$M[i,0]=\sum_{k=1}^{i}\sigma(s_1[i],-)\quad 1\leq i\leq m \tag{2.9}$$

$$M[0,j]=\sum_{k=1}^{j}\sigma(-,s_2[j])\ (1\leq j\leq n) \tag{2.10}$$

对其前缀子序列 $s_1[1...i]$和 $s_2[1...j]$进行分析，要得到最优比对得分 M[i,j]可能有三种情况：(1) $s_1[i]$与 $s_2[j]$ 进行比对的得分与 $s_1[1...i-1]$和 $s_2[1...j-1]$ 两子序列的最优比对得分 M[i-1,j-1]的总和；(2) $s_1[i]$和一个空位“-”进行比对的得分与 $s_1[1...i-1]$和 $s_2[1...j]$两子序列的最优比对得分 M[i-1,j]的总和；(3) $s_2[j]$和一个空位“-”的进行比对的得分与 $s_1[1...i]$和 $s_2[1...j-1]$ 两子序列的最优比对得分 M[i,j-1]的总和。其比对得分来源如图 2.4 所示。

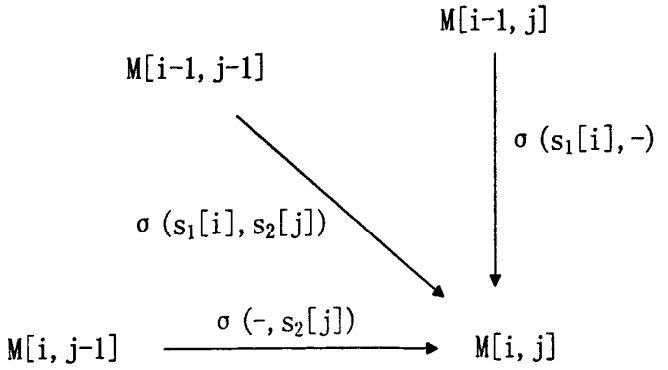

图 2.4 矩阵元素 M[i,j]的来源

由此得到递归关系式：

$$M[i,j]=\max\begin{cases}M[i-1,j-1]+\sigma(s_1[i],s_2[j])\\ M[i-1,j]+\sigma(s_1[i],-)\\ M[i,j-1]+\sigma(-,s_2[j])\end{cases}\quad(1\le i\le m,1\le j\le n)\qquad(2.11)$$

使用上式按照从左至右、从上至下的顺序依次计算矩阵中每个元素的值。当计算到最后一行最后一列的位置，即元素 M[m,n]时，就得到了两序列 s1 和 s2 的最优比对得分。

得到最优比对得分后，采用回溯方法来构造比对结果。即从最优比对得分的位置开始，即矩阵的最后一行最后一列的位置，沿着得到该值的路径进行回溯，直到到达第 0 行第 0 列才停止。此时，回溯过程中经过的交叉点对应的序列就是比对结果。

使用动态规划算法进行序列比对时，需要对大小为(m+1)×(n+1)的二维数组中的每个元素进行计算，其时间复杂度为 O(mn)，空间复杂度为 O(mn)；回溯过程是从数组的右下方走回数组的左上方，经过(m+n)个元素，所以其时间复杂度为 O(m+n)，存储空间没有额外的开销。因此，基本动态规划算法解决双序列比对问题占用的时间和空间都较大，所以人们提出了各种各样的改进算法。

2.2.2.3 多序列比对

多序列比对是双序列比对的一般性推广，即把比对问题从两条序列推广到多条序列上。因此，解决双序列比对问题的方法同样适用于多序列比对，只是比对的个数有所增加，使问题变得更加复杂。Murata 在三条序列的比对中成功地应用了动态规划算法[50]，但是由于序列比对时运行的时间很长、需要的空间很大，所以想进一步推广到三条以上的序列比对中几乎是不可行的。一般情况下，多序列比对很少采用动态规划算法来完成，基本上使用的是启发式算法、随机算法和分治算法等。其中，启发式算法的种类很多，如星比对算法、渐进式比对算法、迭代细化方法等。

1、星比对算法

星比对算法是一种快速的用于求解多序列比对问题的启发式方法[51]。它需要寻找一个中心序列，比对的结果是通过中心序列与其它序列的比对建立的。星比对算法遵循一个“一旦为空格，始终为空格”的法则，即在比对过程中，需要把空格不断地加入到中心序列中从而使得中心序列和比对的序列达到最大匹配数。加入到中心序列中的空格是不能够移出的，并且始终保留在中心序列中，直到中心序列与所有序列都比对完毕。具体算法描述如下：

步骤 1：对于一组含有 K 条序列的集合 Ω 首先找出序列 $S_t(S_t \in \Omega)$，使得 $\sum_{i \neq t} Score(S_t, S_i)$ 的值最大，令 $A = \{S_t\}$；

步骤 2：逐次地往 A 中添加 $S_i(S_i \in \Omega - \{S_t\})$，并使 S_i 与 S_t 的比对的值最大；

假设 A 中已经添加了 $S_1, S_2, ..., S_{i-1}$，由于 A 中的每个序列在与 S_t 进行比对的过程中需要加入空格，故此时 $A = \{S_1', S_2', ..., S_{i-1}', S_t'\}$。按照两条序列比对的动态规划算法比较 S_t' 和 S_i，分别产生新的序列 S_t'' 和 S_i'，再按照 S_t'' 中添加空格的位置调节序列 $\{S_1', S_2', ..., S_{i-1}'\}$ 成 $\{S_1'', S_2'', ..., S_{i-1}''\}$，并用 S_t'' 替换 S_t'，最后得到的比对即星比对。

在星比对过程中，首先必须确认一个中心序列。有两种方法可以实现选择中心序列，一种是逐个测试，择优而取。另一种方法是计算全部优化比对，选择使 $\sum_{i \neq c} Sim(S_c, S_i)$ 最大的序列为中心序列。其中 $Sim()$ 为两个序列的相似性函数，如 $Sim(S_c, S_i)$ 表示序列 S_i 与序列 S_c 的相似程度，可按照下列计算公式确定：

$$Sim(S[1...i], T[1...j]) = \max \begin{cases} Sim(S[1...i], T[1...j-1]) + \sigma(-, T[j]) \\ Sim(S[1...i-1], T[1...j-1]) + \sigma(S[i], T[i]) \\ Sim(S[1...i-1], T[1...j]) + \sigma(S[i], -) \end{cases} \tag{2.12}$$

$$\sigma(i, j) = \begin{cases} +1 & S[i] = T[j] \\ -1 & S[i] \neq T[j] \\ -2 & T[j] = - \\ -2 & S[i] = - \end{cases} \tag{2.13}$$

2、渐进式比对算法

另一种简单且有效的启发式算法就是渐进式比对算法。渐进式比对算法的基本思想是采用动态规划算法迭代地对两序列进行比对。即先比对两个序列，再把新的序列添加进来，直到所有序列都加入进来为止。需要注意的是，不同的比对结果可能是不同的添加顺序造成的。因此，渐进式比对算法的关键是如何确定序列比对的顺序。一般情况下，是从最相似的两个序列开始比对，因为这样的比对的结果人们最有信心，然后由近到远比对完所有的序列。

渐进式比对算法主要由三个步骤完成：(1) 计算距离矩阵；(2) 构建指导树；(3) 根

据构建的指导树把序列逐渐加入进来进行比对。

使用最广泛的渐进式比对程序是 ClustalW。Higgins 和 Sharp 最早编写程序实现了渐进式比对算法，并成功完成了 Clustal 软件包的开发，于 1989 年推出了软件包的 ClustalV 版本。Thompson 等人 1994 年对 ClustalV 进行了改进，开发出了现在广泛使用的 ClustalW。由于 ClustalW 是命令式程序，为了让其操作更为简单，1997 年推出了基于窗口的 ClustalX。

ClustalW 算法给出了一套动态选择比对参数的方案，主要是解决比对过程中参数的选择问题。通常情况下是采用计分矩阵和反射空位罚分来解决比对参数的选择问题，并希望能够设置有效的参数使比对结果达到预期的效果。如果选用的是相似性很大的序列，所设置的参数确实能够达到预期的效果。这是因为在所有的计分矩阵中相同的碱基都会给出最大的分值，所以无论使用何种计分矩阵都能找到近似正确的比对结果。但是对于相似程度很低的序列，计分矩阵的选择就尤为重要，不同的分值可能会产生不同的结果。另外，对于相似程度很高的序列，由于比对结果中插入的空格较少，空位罚分几乎不影响比对结果。当相似程度逐渐降低，空位罚分的精确确定对比对结果也影响很大。ClustalW 采用了启发式方法来选择参数，空位罚分的设置会根据具体的碱基和位置而动态改变，并且随着比对的进行会选择不同的计分矩阵，从而得到更精确的比对结果。

对于进化分布比较均匀的序列，ClustalW 能够快速有效的得到比对结果，但对于分布不均匀或差异性较大的序列，ClustalW 的准确性还有待提高。

2.3 分子进化

2.3.1 分子进化简介

在现代分子进化研究中，根据现有生物基因或物种多样性来构建生物的进化史是一个非常重要的问题。一个可靠的系统发生的推断，将揭示出有关生物进化过程的顺序，有助于我们了解生物进化的历史和进化机制。根据核酸和蛋白质的序列信息，可以推断物种之间的系统发生关系。其基本原理是：从一条序列转变为另一条序列所需要的变换越多，那么这两条序列的相关性就越小，从共同祖先分歧的时间就越早，进化距离就越大；相反，两个序列越相似，那么它们之间的进化距离就可能越小。依据此基本原理，根据推断出的物种的进化关系，用系统进化树的形式表现出来，这就是分子系统学的首要任务。

2.3.2 系统进化树

系统进化树通常是一棵二叉树，分有根和无根两种，前者称为有根树，后者称为无根树。物种或基因的进化顺序可以通过有根树反映出来，而如果只需要清楚物种或基因的分类关系则一般采用无根树。无论是有根树还是无根树，树的分枝式样称为其拓扑结构。如果有 4 个物种（1、2、3 和 4），则有 15 种可能的有根树拓扑结构和 3 种无根树拓扑结构，如图 2.5 和图 2.6 所示。拓扑结构数随着物种的数目增加而迅速增加。

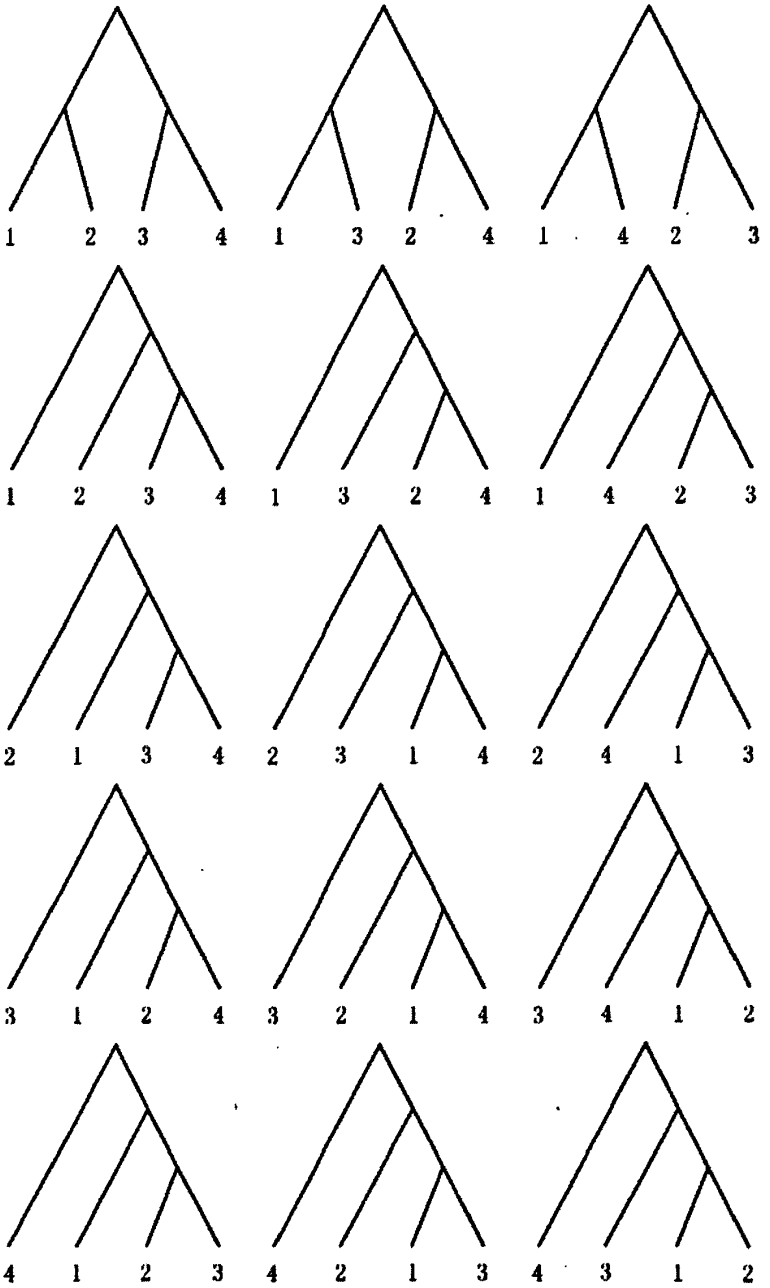

图 2.5 包含 4 个物种的有根树

图 2.6 包含 4 个物种的无根树

假设一棵有根二叉树由 m 个物种或基因构成，其拓扑结构数 N 的计算公式为[52]：

$$N = 1 \times 3 \times 5 \ldots (2m-3) = [(2m-3)!]/2^{m-2}(m-2)! \tag{2.14}$$

假设一棵无根二叉树由 m 个物种或基因构成，其拓扑结构数 N 的计算公式为[53]：

$$T = \prod_{i=3}^{n}(2i-5) \tag{2.15}$$

大多数情况下，拓扑结构中的大部分结构明显表现出不可能的进化关系和其他生物信息，因而很容易被排除。但当 m 的值很大时，要找出反映真实的进化关系的拓扑结构也是一项非常困难的任务。

2.3.3 构建系统进化树

构建进化树的主要步骤有括：（1）获取建树所需的序列数据；（2）建立数据模型，进行序列比对，获取比对结果；（3）选择建树方法构建系统进化树。距离矩阵法、最大简约法和最大似然法是构建进化树的常用方法。多序列比对是距离矩阵法的前提条件，通过进化距离模型，将多序列比对结果转换为反映序列差异程度的距离矩阵。然后在此基础上，设定优化原则和假设条件，选择合适的建树方法构建系统进化树。而最大简约方法和最大似然法都不需要计算距离矩阵，而是将碱基或氨基酸顺序作为基础来构建系统进化树的。通常情况下，距离矩阵法的计算量要远远小于另外两种方法。使用最大简约法构建进化树时需要对每一种拓扑结构上的每个序列的每个位点的替代情况进行计算，如果该拓扑结构中全部位点的替代数之和最小，则认为该结构为最优进化树。一般情况下，最大简约法可能会得到多个最优进化树，而这些最优进化树的全部位点的替代数之和都为最小值。由于没有其他生物进化信息可供参考，任何一棵进化树都可能反映真实情况，所以需要通过其他方法来确保其结果的唯一性。最大似然法是假设物种在发生进化之后，子孙物种之间是相互独立的，不会再相互影响。根据这个假设，使用最大似然法构建进化树的主要思路是首先对每种拓扑结构计算其系统进化树的似然值，其值为所有位点似然值的乘积；然后对每种拓扑结构的似然值进行比对，其最大者为最优进化树。

2.4 小结

本章首先介绍了生物信息学的基本概念，包括核酸、蛋白质、遗传信息的传递与表达和变异等。并描述了生物信息学中的序列比对问题，详细介绍了多序列比对所涉及的基本问题，如：多序列比对的数学描述、空位罚分、计分矩阵，序列比对评价标准。本章接着介绍了双序列比对算法和多序列比对算法。对最为经典的双序列比对算法：点阵图法和动态规划算法进行了详细的介绍。并重点介绍了基于启发式算法的星比对算法和基于渐进比对方法的 ClustalW 算法。最后在分子进化方面，对什么是系统进化树、如何构建系统进化树进行了简单的描述。

第 3 章 基于模式匹配的 DNA 多序列比对算法

目前，多序列比对大多是采用启发式算法或组合优化算法求得其近似最优解来解决。本文通过分析基于关键字树的 DNA 多序列星比对算法[42]，并在其基础上提出了基于模式匹配的 DNA 多序列比对算法。

3.1 模式匹配原理及应用

3.1.1 模式匹配概述

模式匹配是指在文本序列 $Text = t[1]t[2]..t[n]$ 中检索某个特定模式子串 $Pattern = P[1]P[2]...P[m]$ 的所有出现位置。这里，定义Σ为有限字符集，$t[i] \in \Sigma | 1 \le i \le n$，$P[i] \in \Sigma | 1 \le i \le m$，n 为 Text 的长度，m 为 Pattern 的长度。在模式匹配过程中，如果 Text 中存在子串匹配 Pattern 中的某个模式，则匹配成功并给出该子串在 Text 中的位置，否则匹配失败[54]。

模式匹配技术已经被广泛应用于不同的领域中，如入侵检测、内容过滤防火墙以及计算生物学的 DNA 序列匹配等。随着互联网络技术与生物信息的迅速发展，近年来越来越多的新的应用需求对模式匹配技术提出了新的挑战。

3.1.2 模式匹配分类

1、按功能分类

模式匹配根据功能可分为精确模式匹配、近似模式匹配和正则表达式匹配三类。

精确模式匹配是指在文本序列中找出的子串完全匹配指定的模式串，并返回这些子串的位置。精确模式匹配一般使用一个固定长度的窗口来搜索文本，检测窗口内的文本是否匹配，然后根据匹配情况尽可能地将窗口向右移动，直到将整个文本搜索完成。根据检索方式的不同，精确模式匹配可以分为前缀模式、后缀模式和子串模式三种[54]。他们的主要区别在窗口的移动策略和如何匹配文本上。

近似模式匹配是指给定一个相似程度的度量标准，在文本序列中找出的子串在一定范围内匹配模式串，并返回这些子串的位置。近似模式匹配允许文本序列中的子串和模式串之间存在一定的误差，误差是根据给定的相似程度度量标准来衡量。一般有两种匹配方法允许 k-差别的近似模式匹配和允许 k-误配的近似串匹配。两种方法使用不同的计算误差的方法，前者使用的是编辑距离，后者使用的是汉明距离。

正则表达式匹配使用正则表达式来定义文本序列中匹配的模式，在文本序列中查找匹配的子串，并返回子串的位置。正则表达式匹配的实现基于 NFA 和 DFA 两种方式。其中，DFA 占用内存空间太大，一般采用转换压缩、状态压缩和字母表压缩三种方式进行空间压缩[54]。

2、按一次能够匹配的模式数量分类

模式匹配根据一次能够匹配的模式数量可分为单模式匹配和多模式匹配。单模式匹配只有一个匹配的模式串，在查找文本序列时可以采用依次读入每个字符或滑动窗口的方法，检验输入的子串是否匹配该模式串，并返回其出现的位置。经典的单模式匹配算法包括 BF 算法[55]、KMP 算法[56]、BM 算法[57]等。多模式匹配存在着多个匹配的模式串，在查找文本序列时检验输入串是否匹配其中一个或多个模式串，返回其出现的位置。经典的多模式串匹配算法包括 Aho-Corasick 算法[58]等。

3.1.3 模式匹配常用算法

目前，学者们提出了许多有效的模式匹配算法，并针对模式匹配的时间和空间进行了有效的处理。

(1) BF（Brute-Force）算法

单模式匹配算法中最早出现并且最简单的就是 BF 算法。其处理过程是将文本序列和模式串从左至右进行对齐，并逐个匹配文本序列中的字符和模式串中的是否完全相同，如果相同就匹配成功，如果某个字符不相同就匹配失败。这时将模式串右移一位，重新开始从左至右进行逐个字符的匹配，直到匹配到文本序列的末尾。该算法是最简单的算法，容易处理，并且不需要额外的存储空间，但处理过程存在回溯，效率很低。

(2) Aho-Corasick 算法

Aho-Corasick 算法是由 A.V.Aho 和 M.J.Corasick 提出的，它是经典的多模式匹配算法之一。该算法首先扫描模式串集合，将集合中提供的模式串构建一棵模式树，然后对文本序列进行扫描，只需扫描一次就能找出所有匹配的模式串，匹配的效率很高。

Aho-Corasick 算法根据处理过程可以分为构树和匹配两个阶段。

① 构树阶段

构树阶段的任务主要是扫描所有的模式串构建一棵模式树。模式树 M 是一个六元组，见式(3.1)：

$$M=(Q,\Sigma,g,f,q_0,p) \tag{3.1}$$

其中，Q表示模式树上的节点集合。

Σ表示有限字符集，即构成模式串的字符集合。

g 表示转移函数，即匹配成功时应该如何沿着模式树进行转移。

f 表示失效函数，即匹配不成功时应该如何转移。

q_0 表示初始状态，即根节点，其标志符为 0，且 $q_0 \in Q$。

p 表示输出函数，即输出所有匹配的模式集合。

构树阶段主要有 3 个函数：转移函数 g()，失效函数 f()和输出函数 p()。

转移函数 g(s,c)表示匹配成功时沿着怎样的路径转移。假设 u、v 分别表示模式树上的两个节点，如果(u，v)边为 c，则 g(u,c)=v；如果 u 为根节点，并且不存在值为 c 的边，则 g(0,c)=0。转移函数的构造过程需要扫描整个模式串集合。首先从模式串集合中依次取出每个字符，从根节点开始，检验取出的字符和模式树中当前状态中的字符是否匹配。如果从当前状态出发存在着到下一个状态经过该字符，则将下一个状态设为当前状态。否则从当前状态出发添加一个大于当前状态标号的新状态，当前状态到新状态经过该字符，并将新加入的状态设为当前状态。当集合中的所有模式串都处理完毕后，再添加一条从根节点到根节点的自返线，处理那些不能从 0 状态开始的字符集。

失效函数 f(s)表示匹配不成功时应该如何转移，s 为当前状态。在检验取出的字符和模式树中当前状态的字符是否匹配时，当前状态不存在与该字符相匹配的转换，这时匹配过程将转到失效函数 f(s)，通过模式树查找除该分支外其他分支是否有匹配的字符。在构造失效函数时，所有从根节点出发的状态的失效函数值为 0，即回到根节点。如果当前状态为 s，其父状态为 r，状态 r 经过字符 a 到达状态 s，则有 g(r,a)=s。如果考虑 r 的失效函数 f(r)，假设 g(f(r),a)存在，那么 f(r)=g(f(r),a)，假设 g(f(r),a)不存在，则继续追溯 f(r)的失效函数 f(f(r))。直至追溯到状态 s' 使得 $g(f(s'),a)$ 存在，那么 $f(s)=g(f(s'),a)$。

输出函数 p(s)表示输出集合中所有匹配的模式串，s 为当前状态。如果没有模式串匹配成功，则 p(s)为空。输出函数的处理分两种情况，当使用转移函数处理完整个模式串时，将该模式串以及出现的位置加入到 p(s)中；当使用失效函数处理模式匹配时，通过失效函数转移到的模式也要加入到 p(s)中。

② 匹配阶段

匹配阶段的主要任务是利用已经构造好的模式树对文本序列进行一次性扫描，找出匹配的模式串并返回其位置。匹配阶段的操作过程具体描述为：从模式树的根节点即 0 状态出发，逐个取出文本序列的每一个字符，并根据转移函数 g()或者失效函数 f()进入到下一个状态。如果匹配成功则将当前匹配的模式和出现的位置保存到 p()中。以此类推，直到取出文本序列的最后一个字符。

Aho-Corasick 算法的匹配过程非常简单，可以根据匹配失效时已匹配的字符来确定移动的位置。假设文本序列的长度为 n，模式串集合的总长度为 m。在构树阶段，

Aho-Corasick 算法需要对模式串集合中的每个字符进行扫描，所以构树阶段的时间复杂度为 O(m)。在匹配阶段，Aho-Corasick 算法需要对整个文本序列中的每个字符进行扫描，所以匹配阶段的时间复杂度为 O(n)。这与模式串的个数以及每个模式串的长度是没有关系的。无论该模式串是否出现在文本序列中，都需要读入文本序列中的每个字符，所以无论是最好情况还是最坏情况，其时间复杂度都是 O(n)。由此可知，Aho-Corasick 算法总的时间复杂度为 O(M+n)，说明该算法只与长度有关，即文本序列的长度和模式串的总长度，而与文本序列中有些什么字符，是否和模式串中的一致是没有关系的。

3.1.4 模式匹配技术在计算生物学中的应用

生物信息学，这一新型的交叉学科随着人类基因组测序工程的完成以及大型序列数据库的建立不断发展。生物信息学中最基本的问题就是序列比对，人们采用了不同的方法来处理序列比对，其中最常用的方法就是把基因序列看成是由有限字符集中的字符构成的文本字符串，然后进行文本字符串的比对。所以许多生物信息学的相关问题通过转换将基因序列转换为文本序列之后，就可以使用模式匹配技术来解决[59]。并且模式匹配技术的处理效率也直接或间接地影响到生物信息数据处理和分析的效率。

3.2 基于模式匹配的多序列比对算法引入

目前大多是应用启发式算法或组合优化算法求得 DNA 多序列比对近似最优解。其中，星比对算法就是启发式算法中的一种。由于原始星比对算法的执行时间太长，无法满足实际需要，文献[42]对原始星比对算法进行改进，提出了基于关键字树的 DNA 多序列星比对算法。主要思想是依次将每个序列都分割成等长的子串集合，为这个集合构建一棵模式树。除该序列的其他序列使用 Aho-Corasick 算法对该模式树进行模式匹配，并记录所有模式串出现次数的总和。将出现次数总和值最大的序列定为中心序列，然后根据星比对算法中“一旦为空格，始终为空格”的思想，将其他序列与中心序列的未匹配部分进行多序列比对。文献[42]从两个方面的原始的星比对算法进行了改进。一方面在寻找中心序列时，引入了分割的思想和 Aho-Corasick 算法，大大降低了其运行时间；另一方面在中心序列与其他序列比对时，已经匹配的部分不需要再进行比对，大大提高了算法的执行效率。

但由于文献[42]提出的算法的适用范围是相似程度很高的 DNA 序列，为了提高该算法的比对范围，进一步降低其期望运行时间，本文提出了基于模式匹配的多序列比对算法。该算法同样是基于星比对算法的，所以仍然包括寻找中心序列和两两比对两个步骤。与文献[42]不同之处在于本文通过建立一棵公共的模式树，并用二维数组记录每个序列

搜索该树得到的模式号，通过对二维数组中的模式号的逐级筛选来寻找中心序列。

另外，在比对过程中由于两个序列之间的匹配部分已经被记录下来，所以只需要使用动态规划法比对那些不匹配的部分。结果证明该算法是可行的DNA多序列比对方法。

3.3 算法描述

基于模式匹配的多序列比对算法仍然是基于星比对算法的，分为寻找中心序列和两两比对两个主要步骤。要寻找中心序列，设需要进行比对的多个序列分别是$P_1,P_2,...,P_n$。具体操作如下：

(1) 将每个序列$P_i(i=1,2,...,n)$等分成长度为r的子序列构成子序列集合L，$r=\lfloor l/k \rfloor$(l取值$len(P_1)$，k取值$O(l/\log l)$)。将集合L中的所有子序列构成一棵公共的模式树T。

(2) 对于公共的模式树T，每个序列$P_i(i=1,2,...,n)$利用Aho-Corasick算法搜索T，并记录每个序列$P_i(i=1,2,...,n)$匹配的模式号和该模式在当前序列中出现的位置。这里使用一个三维数组进行记录，三维数组的第一维对应一个序列，第二维包括2个一维数组，一个数组存储匹配的模式号，另一个数组存储该模式号在序列中出现的位置。

(3) 统计每个序列$P_i(i=1,2,...,n)$匹配的模式号，这里只考虑搜索T时通过非失效链接得到的模式号，并使用一个二维数组进行统计。二维数组的第0行按顺序记录匹配成功的模式号的值。从第1行开始，每一行对应一个序列，该行上的每一列对应模式号在该序列中出现的次数。

(4) 依次找出匹配次数最多的那个模式号，将没有匹配该模式号的序列从查找中心序列的队列中去除，最后剩下的序列将是中心序列P_c。

(5) 得到中心序列后，将P_c与$P_i(i=1,2,...,n且i\neq c)$依次进行比对。由于在步骤(4)中已经记录了P_c和P_i匹配的子串位置，然后使用动态规划算法将P_c和P_i未匹配的子串进行比对。并且记录需要在P_c和P_i中插入空格的位置S_{ci}和S_i。

(6) 依次比对完P_c和$P_i(i=1,2,...,n且i\neq c)$后，对所有需要插入空格的位置$S_{ci}(i\neq c)$汇总，汇总后的位置记为S_c，分别比较S_c和$S_{ci}(i=1,2,...,n且i\neq c)$，在$S_i$中加入新汇入的空格位置。分别根据$S_i$中记录的空格位置将空格插入到$P_i$中并输出多序列比对的结果。

3.4 算法复杂性描述

假设n个序列$P_1,P_2,...,P_n$的长度均为m，将每个序列等分成k段建立公共的模式树所花费的时间是$O(nm)$。最坏情况下，每个序列都不存在相同的模式，所以n个序列共有模式号为$O(nk)$，在寻找中心序列时每次对全部模式号的匹配个数进行累加和比较，花费时间为$O(n^2k)$。同样考虑最坏情况，每次只去掉一个序列，共需要花费时间为

$O(n^3k)$。所以最坏情况下，找出中心序列所花费的时间为$O(nm)+O(n^3k)$。最好情况下，每个序列都存在相同的模式，所以 n 个序列共有模式号为$O(k)$，在寻找中心序列时对所有模式号的匹配个数进行累加和比较，并且一次性找出中心序列，所以花费时间为$O(nk)$。

找出中心序列以后，还需要对中心序列和其他序列进行 n-1 次双序列比对，运行时间为$O(nm^2)$。因为在中心序列的查找中使用 Aho-Corasick 算法已经记录了完全匹配的子串的位置，所以只需要对不匹配的部分进行序列比对。如果两个序列非常相似，要比对的子序列长度远远小于 m，也会大大降低运行时间。当相似程度很高的多个序列进行比对时，与寻找中心序列所用的时间进行比较，序列比对的时间几乎可以忽略不计。

在中心序列与其他序列两两比对完成之后需要对插入空格的位置进行汇总并输出比对的结果。汇总空格和输出比对结果时都需要对每个序列的每个字符位置进行扫描一次，因此其运行时间分别都是 O(nm)。

因此，本算法的最坏时间复杂度是 $O(nm)+O(n^3k)+O(nm^2)$，由于 n 远远小于 m，k 也是远小于 m 的值，可以认为最坏时间复杂度为 $O(nm^2)$，好于原始星比对算法的 $O(n^2m^2)$。与文献[42]提出的“基于关键字树的 DNA 多序列星比对算法”的时间复杂度 $O(nm)+O(n^2m)+O(nm^2)$相比较，主要区别在于 $O(n^2m)$。由于 k 取值 O(m/log m)，只有 n 远远小于 m，本算法才会优于文献[42]提出的改进的星比对算法。本算法的最好时间复杂度是 $O(nk)+O(nm^2)+O(nm)$，所以当序列相似度很高时，本算法也会优于基于关键字树的 DNA 多序列星比对算法。

3.5 实验结果对比分析

本实验将基于模式匹配的多序列比对算法和原始星比对算法、基于关键字树的 DNA 多序列星比对算法进行比较。实验 PC 的 CPU 为 Pentium(R) Dual-Core T4200 2.00GHz，内存为 2.00GB，操作系统为 Windows7。基于模式匹配的多序列比对算法使用 C#语言在 Vistual Studio 2008 上实现。本次实验主要是为了比较 3 种算法的期望时间复杂度和比对的效率，所以使用 SPS 值和程序平均运行时间作为评价测试结果的标准。

3.5.1 实验结果

实验中，选取了三组数据进行测试，并且对每个测试数据都连续运行程序 20 次，最终取其运行时间的平均值，并对基于模式匹配的多序列比对算法、基于关键字树的 DNA 多序列星比对算法和原始星比对算法在程序平均运行时间和 SPS 值进行了比较。

第一组实验通过对 6 种西藏部分地区豆科植物的根瘤菌多序列比对，完成 DNA 同

源性分析。实验数据来自于 NCBI(http://www.ncbi.nm.nih.gov)上的核酸库以及文献[60]，其序列名称、序列号及序列长度见表 3.1 所示。序列的条数为 6，序列的平均长度为 1428。该实验数据程序运行时间的统计如表 3.2 所示。

表 3.1 6 种西藏根瘤菌 RDNA 全序列

名称	序列号	长度（bp）
Rhizobium mongolense	U89817	1421
Rhizobium gallicum	U86343	1474
Rhizobium yanglingense	AF003375	1357
Rhizobium leguminosarum	U29386	1423
Sinorhizobium meliloti	X67222	1428
Sinorhizobium morelense	AY024335	1436

表 3.2 6 种西藏根瘤菌 RDNA 全序列比对运行时间统计（单位：ms）

次数	原始的星比对算法	基于关键字树的星比对算法	基于模式匹配的多序列比对算法
1	2228	37	55
2	2199	37	52
3	2206	38	53
4	2207	36	53
5	2209	36	52
6	2201	38	58
7	2206	35	54
8	2203	35	52
9	2200	37	51
10	2197	39	48
11	2205	40	54
12	2207	39	49
13	2200	37	52
14	2200	38	51
15	2200	36	52
16	2203	36	60
17	2199	38	52
18	2198	36	49
19	2206	36	52
20	2198	35	50
平均值	2203.60	36.95	52.45

统计 6 种西藏根瘤菌 RDNA 全序列的碱基对匹配情况如表 3.3 所示。

表 3.3 6 种西藏根瘤菌 RDNA 全序列碱基对匹配情况对比（单位：个）

	原始的星比对算法	基于关键字树的星比对算法	基于模式匹配的多序列比对算法
Rhizobium mongolense	6878	6798	6878
Rhizobium gallicum	6904	6806	6905
Rhizobium yanglingense	6645	6646	6644
Rhizobium leguminosarum	6819	6716	6820
Sinorhizobium meliloti	6784	6697	6784
Sinorhizobium morelense	6854	6731	6853

比较三种算法的 SPS 分值及程序平均运行时间如表 3.4 所示。

表 3.4 6 种西藏根瘤菌 RDNA 全序列比对结果情况对比

	原始的星比对算法	基于关键字树的星比对算法	基于模式匹配的多序列比对算法
SPS 分值	0.916	0.886	0.916
程序平均运行时间（ms）	2203.60	36.95	52.45

第二组实验通过对 11 个物种的 β-球蛋白基因的第一个外显子进行多序列比对。实验数据来自于 NCBI（http://www.ncbi.nm.nih.gov）上的核酸库以及文献[25]，其序列名称和序列号见表 3.5 所示。序列的条数为 11，序列的平均长度为 92。该实验数据程序运行时间的统计如表 3.6 所示。

表 3.5 11 个物种的 β-球蛋白基因第一个外显子

名称	序列号	长度（bp）	第一个外显子
Human	U01317	92	62187-62278
Goat	M15387	86	279-364
Gallus	V00409	86	465-556
Rat	X06701	94	310-401
Mouse	V00722	92	275-367
Chimpanzee	X02345	105	4189-4293
Bovine	X00376	86	278-363
Gorilla	X61109	93	4538-4630
Opossum	J03643	92	467-558
Lemur	M15734	92	154-245
Rabbit	V00882	90	277-368

表 3.6 11 个物种的 β-球蛋白基因第一个外显子比对运行时间统计（单位：ms）

次数	原始的星比对算法	基于关键字树的星比对算法	基于模式匹配的多序列比对算法
1	35	10	4
2	36	10	5
3	36	9	5
4	35	10	5
5	36	10	6
6	36	10	4
7	36	10	6
8	37	9	4
9	37	10	6
10	36	10	5
11	36	10	5
12	36	10	5
13	35	10	6
14	36	9	4
15	36	10	5
16	36	10	5
17	36	10	5
18	37	9	5
19	36	10	5
20	36	10	5
平均值	36.00	9.80	5.00

统计 11 个物种的 β-球蛋白基因第一个外显子的碱基对匹配情况如表 3.7 所示。

表 3.7 11 个物种的 β-球蛋白基因第一个外显子碱基对匹配情况对比（单位：个）

	原始的星比对算法	基于关键字树的星比对算法	基于模式匹配的多序列比对算法
Human	782	777	782
Goat	735	730	735
Gallus	735	730	735
Rat	722	717	722
Mouse	687	682	687
Chimpanzee	790	791	790
Bovine	741	736	741
Gorilla	790	785	790
Opossum	660	618	660
Lemur	641	636	641
Rabbit	729	726	729

比较三种算法的 SPS 分值及程序平均运行时间如表 3.8 所示。

表 3.8 11 个物种的 β-球蛋白基因第一个外显子的比对结果情况对比

	原始的星比对算法	基于关键字树的星比对算法	基于模式匹配的多序列比对算法
SPS	0.681	0.674	0.681
程序平均运行时间（ms）	36.00	9.80	5.00

第三组实验通过对相似性很高的一组 8 种 H5N1 型禽流感病毒的 HA 片段基因序列进行多序列比对。实验数据来自于 NCBI（http://www.ncbi.nm.nih.gov）上的核酸库以及文献[26]，其序列名称和序列号见表 3.9 所示。序列的平均长度为 1699，序列条数为 8。该实验数据程序运行时间的统计如表 3.10 所示。

表 3.9 8 种 H5N1 型禽流感病毒 HA 片段

名称	序列号	长度（bp）
DG12	AY585361	1708
DG40	AY585374	1708
DG1681	DQ320886	1698
DG1793	DQ320887	1698
DH1265	DQ320911	1695
DH1608	DQ320912	1695
CY447	DQ095624	1698
CH18	DQ320925	1695

表 3.10 8 种 H5N1 型禽流感病毒 HA 片段序列的比对运行时间统计（单位：ms）

次数	原始的星比对算法	基于关键字树的星比对算法	基于模式匹配的多序列比对算法
1	5587	413	47
2	5569	405	44
3	5563	418	45
4	5566	403	45
5	5584	438	53
6	5567	403	52
7	5571	413	51
8	5563	419	46
9	5708	412	45
10	5570	426	46
11	5570	401	46
12	5571	401	46

续表 3.10

次数	原始的星比对算法	基于关键字树的星比对算法	基于模式匹配的多序列比对算法
13	5567	416	43
14	5558	405	47
15	5591	414	47
16	5799	399	50
17	5575	413	54
18	5581	404	53
19	5565	409	46
20	5596	407	47
平均值	5591.05	410.95	47.65

统计 8 种 H5N1 型禽流感病毒 HA 片段序列的碱基对匹配情况如表 3.11 所示。

表 3.11 8 种 H5N1 型禽流感病毒 HA 片段序列的碱基对匹配情况对比（单位：个）

	原始的星比对算法	基于关键字树的星比对算法	基于模式匹配的多序列比对算法
DG12	11403	11399	11407
DG40	11542	11534	11542
DG1681	11420	11419	11420
DG1793	11420	11419	11420
DH1265	11493	11494	11494
DH1608	11479	11480	11480
CY447	11407	11402	11408
CH18	11486	11487	11487

比较三种算法的 SPS 分值及程序平均运行时间如表 3.12 所示。

表 3.12 8 种 H5N1 型禽流感病毒 HA 片段序列的比对结果情况对比

	原始的星比对算法	基于关键字树的星比对算法	基于模式匹配的多序列比对算法
SPS	0.958	0.956	0.958
程序平均运行时间（ms）	5591.05	410.95	47.65

3.5.2 实验结果分析

通过第一组实验结果可知，在运行时间上本算法是原始星比对算法的 2.36%，而基于关键字树的星比对算法的运行时间是原始星比对算法所用时间的 1.68%，而本算法的 SPS 分值高于基于关键字树的星比对算法。由于本算法在寻找中心序列时采用的是逐级

去掉希望最小的序列，最后剩下的序列所包含的公共的子串为最多，成为中心序列的几率也就越大。虽然在寻找中心序列时所花费的时间相对基于关键字树的星比对算法可能有所增加，但本算法得到的结果更接近最优解，影响生物实验的结论的可能性更小。

在第二组和第三组实验中，本算法的碱基对匹配个数也都高于基于关键字树的星比对算法，与原始星比对算法更为接近。在运行时间方面，寻找中心序列时统计每个序列共有的模式号的个数接近 k 值，所以可以认为寻找中心序列的运行时间为 $O(n^2k)$。而基于关键字树的星比对算法的寻找中心序列运行时间为 $O(n^2m)$，k 是待比较的序列所分割的段数，远小于 m，所以 $O(n^2k)$也远小于 $O(n^2m)$。在建立好关键字树后，本算法和基于关键字树的星比对算法一样，将完全匹配的子串位置存储下来。所以在其他序列与中心序列逐个进行比对的过程中，对于完全匹配的子串就不需要再进行比对。另外，其他序列与中心序列采用动态规划算法进行比对，其运行时间与比对的长度呈平方关系。如果相似程度非常高的序列进行比对，匹配的子串个数与 k 越接近，那么在每个序列与中心序列进行比对的长度就越短，花费的时间也就越少。

3.6 小结

本章介绍了一种新的生物序列比对算法——基于模式匹配的 DNA 多序列比对算法。此算法是在模式匹配理论和 Aho-Corasick 搜索算法的基础上，针对基于关键字树的 DNA 多序列比对方法提出来的。

由于此算法是对原始星比对算法的一种改进，所以算法的实现仍然包括两个步骤：寻找中心序列和序列两两比对。本章通过三组实验，比较了原始星比对算法、基于关键字树的 DNA 多序列比对算法和新算法。结果表明：新算法相对于基于关键字树的 DNA 多序列比对算法的准确率更高，相对于原始星比对算法的运算速度更快。当比对的序列相似性很高时，新算法的运算速度也要优于基于关键字树的 DNA 多序列比对算法。

但是此算法在寻找中心序列时还存在不安全的因素。当变异率较大时，也会大大影响到比对结果。此算法还有待改进。

第 4 章 基于模式匹配的 DNA 序列相似性分析

DNA 序列相似性分析是研究序列是否具有同源关系以及通过已知序列与未知序列的相似性分析预测未知序列的结构和功能等。序列相似性分析的方法很多，本章采用序列比对的方法，即使用模式匹配的方法进行多序列比对，并将序列比对的结果使用 Kimura 双参数模型和 Neighbor-joining 方法构建进化树，进行序列相似性分析。

4.1 常用的序列相似性分析方法

由于 DNA 原始序列很长，很难从序列本身获取信息直接进行比较分析，所以国内外学者提出了不同的方法来实现序列的相似性分析。主要的方法有信息理论方法、序列比对方法、统计特征方法、基于图形表示的方法和聚类分析方法。

4.1.1 信息理论方法

信息理论方法是基于 DNA 序列的原始信息的，和任何主观因素是没有关系的。研究结果表明使用一些信息理论方法并没有得到预期的结果，而另一些信息理论方法并不适用于序列相似性分析。一般用于 DNA 序列相似性分析的信息理论的基本方法主要有两种：条件熵[61]和熵估计法。

2001 年，方伟武提出了一种 FDOD（Function of Degree of Disagreement）方法[62]，即一种新型的信息离散性度量方法。该方法以某一长度的子序列概率分布作为序列的特征，并使用 FDOD 函数计算子序列概率分布之间的距离。由于该方法能够有效计算出多个分布之间的距离，所以能够较好地解决多序列比对问题。2009 年，刘芳提出了修正的广义信息距离[26]；即一种新的基于信息离散度的序列差异度量方法。该方法既适用于相似程度很高的序列，也适用于差异性较大的序列。

4.1.2 序列比对方法

序列比对方法是 DNA 序列相似性分析方法中应用最广泛并且最基础的方法，采用这种方法进行相似性分析需要使用者设置相关参数、比对罚分值、插入空位的限制等等得到长度相等的序列比对结果，然后使用序列比对的结果进行相似性分析。本文中将采用这种方法实现。

4.1.3 统计特征方法

生物序列并不是由一些特定字符随机构成的序列，不同的物种，同一物种序列的不同区域对不同的碱基和不同的氨基酸都有不同的偏好。因此，在一定程度上，子串的出现频率或字符间的前后关系都能唯一代表一条序列。

基于特征的统计方法，与传统的基因组特征提取方法不同的地方在于它是不依赖于序列比对的结果的；并且在一定程度上序列片断的统计特征可以反映序列的整体特征；统计的特征来源于序列的基本信息，数据基础丰富；统计的特征应用范围较广，可以适应于序列进化分析，序列相似性分析以及功能片断识别、结构片段识别等等。基于特征的统计方法通常有：WF 特征，BBC 特征[63]，DRA 特征[64]。

4.1.4 基于图形表示的方法

人们最初都是从 DNA 原始序列本身获取生物学信息，并且也非常困难。后来人们发现将数学学科中的几何学引入到序列的表示方法中，能够更加形象直观地表示生物序列。并且将生物序列的图形表示转换成一些数值特征，能够让我们更加理性地分析序列。

基于图形表示的分析方法就是将生物序列转换为几何图形，然后根据图形表示构造矩阵，并从矩阵中提出矩阵相关不变量作为序列的特征值，可以是行和元素平均值、最大特征值等等。最后根据其特征值进行序列的相似性分析。

E.Hamori 和 J.Ruskin 于 1983 年提出 G-曲线[65]，这是最早提出用空间曲线表示 DNA 序列的。由于 G 曲线是一种 5 维空间表示，不具有图形可视化优点。1990 年，Jeffrey 提出 CGR 图表示法[66]，Gates M A、Nandy A、Leong P M 和 Morgenthaler S 提出了二维空间表示法[67~69]。这三种方法都可能出现图形上的交叠现象，即退化现象。

我国张春霆院士提出一种 Z 曲线[66]，这种曲线能够直观的显示和分析 DNA 序列，并为使用几何学方法分析和研究 DNA 序列开辟了新的领域。不足的是 Z 曲线中仍然可能存在回路等退化现象。于是，文献[34,70,71]在图形表示方法的基础上引入了映射规则，文献[72,73]在图形表示方法的基础上突出相邻核苷酸所隐藏的信息，都能够较好地应用于基因序列的相似性分析研究中。

4.1.5 聚类分析方法

数据挖掘的主要任务之一就是聚类分析。聚类分析是根据不同类或簇中的对象数据差异性较大的原则将所需要分析的对象数据分成若干个类或簇的过程。其目的是发现同

一类或簇中的个体共性和不同类或簇中个体的异性，从而发掘对象数据中存在的潜藏的规律。

聚类分析被广泛应用到生物信息学研究的各个领域中。主要的聚类分析技术有：层次聚类、K-均值聚类、模糊 C-均值聚类、自组织图聚类和主成分分析等。通过层次聚类可以直接反映出基因或物种之间的亲缘关系，对构建进化树有很大的帮助。聚类分析技术中最常见的是 K-均值聚类，它的数据聚类结果近似于球状。模糊聚类需要在聚类过程中动态确定聚类的数目。自组织图聚类是用几何图形表示数据聚类情况。

4.2 利用模式匹配的序列相似性分析方法

利用模式匹配的序列相似性分析方法属于 DNA 序列相似性研究中应用最广泛的序列比对方法，该方法基于模式匹配的序列比对结果进行相似性分析。为了使得到的进化树更加准确，我们采用 Kimura 双参数模型和 Neighbor-joining 方法实现。

4.2.1 构建进化树

构建进化树是指根据生物序列相似性分析得到的信息推断出物种间的进化关系，并把这种进化关系用一棵树表示出来，一个物种对应树的一个叶子节点，物种之间的进化距离用树枝长度表示出来。进化树的构建通常可以分为两类，距离矩阵法和基于特征的方法。

采用距离矩阵法构建进化树是指假设序列的条数为 n，序列 i 和序列 j 之间的进化距离是 d_{ij}，就得到一个 n 行 n 列的二维矩阵，即距离矩阵，然后根据距离矩阵构建进化树。采用这种方式构建的进化树具有如下的特点：

1. 一个叶子节点对应一个序列，即这棵树有 n 个叶子节点。
2. 在构建进化树时，假设第 i、j 个叶子节点间的距离为 $\overline{d_{ij}}$，要使 $\sum_{i<j<n}(\overline{d_{ij}}-d_{ij})^2$ 最小。

采用距离矩阵法构建进化树实际上就是根据给定的一组距离数据按照指定的规则抽取出建树的叶子节点，这个规则就是树中对应的路径长度和各叶子节点之间的进化距离之间的误差的平方和具有最小值。

4.2.2 Kimura 模型

距离矩阵法是一种很直接的方法，主要是根据序列之间的进化距离来生成进化树。所以生成的进化树的好坏取决于进化距离度量的质量好坏，而进化距离又取决于进化模

型。所以只有选择正确合适的进化模型才能构建出优质的进化树。在构建进化树时，Jukes-Cantor 模型和 Kimura 模型[74,75]是最常用的进化模型。

我们认为所有物种在生物进化过程中都是由同一个祖先进化而来的，不同的子孙后代是由这一祖先经过不同的进化过程，发生了一系列不同的氨基酸的插入、删除和突变而形成的。DNA 序列的进化距离 d 反映了 DNA 序列间的分歧度，即 DNA 序列间的差异程度。Jukes-Cantor 提出的单参数模型和 Kimura 提出的双参数模型都可以计算出 DNA 序列间的进化距离。Judes 和 Cantor 在同源性的前提条件下，假设碱基替换频率为 μ，则碱基的突变频率是相等的，其值为一个常量 μ / 3，这里的突变指的是一碱基突变成另外三种碱基。Kimura 对突变频率作进一步研究，发现转换和颠换的频率是不相同的。转换，I 型变换，是指突变发生在两种嘧啶碱基之间或两种嘌呤碱基之间，即 A↔G；T↔C。颠换，II 型变换，是指突变发生在一个嘧啶碱基和一个嘌呤碱基之间，即 A↔C；A↔T；G↔C；G↔T。用 a 和 β 表示不同的频率，具体见表 4.1 所示。

表 4.1 Kimura 双参数模型

	A	T	G	C
A	-	β	α	β
T	β	-	β	α
G	α	β	-	β
C	β	α	β	-

Kimura 双参数模型的进化距离计算公式如下：

$$\bar{d} = -\frac{1}{2}\ln[(1-2P_I-2P_{II})\sqrt{1-2P_{II}}] \approx 2(\alpha+2\beta)t \tag{4.3}$$

其中，(a+2β)是单位时间碱基突变的总频率值。

P_I 是 DNA 序列中转换突变的概率。

P_{II} 是 DNA 序列中颠换突变的概率。

P_I 的计算公式如下：

$$P_I = \frac{\text{两个比对序列中I型变换的碱基总数}}{\max(\text{两个序列的总长})} \tag{4.4}$$

P_{II} 的计算公式如下：

$$P_{II} = \frac{\text{两个比对序列中II型变换的碱基总数}}{\max(\text{两个序列的总长})} \tag{4.5}$$

4.2.3 Neighbor-joining 方法

Neighbor-joining 方法是由 Saitou 和 Nei 提出。其主要思想是反复地筛选出节点集合中最相邻的两个点构成一个新的节点加入到集合中，并从集合众去除筛选出来的两个节点，直到集合中的节点都处理完毕。其中，最相邻的两个点是指速率校正距离最小的两个节点。该方法的初始状态是一棵星状树，计算任意两个分类群的速率校正距离，取值最小的两个分类群进行聚类，并得到新的分类群与其他分类群的距离，重复上述过程，直到构成一棵以所有分类群为叶子节点的系统树[29]。

假设由 n 个生物序列（DNA 序列或蛋白质序列）构成的集合 $C=\{S_1,S_2,...,S_n\}$；$|C|$ 表示集合中的元素个数，$|C|=n$；Φ 表示空集；如果 $\Omega=\{C_1,C_2,...,C_n\}$，满足：

$$C=\bigcup_{i=1}^{n}C_i \tag{4.6}$$

其中，$\forall i\neq j,1\leq i,j\leq n,C_i\cap C_j=\Phi$，则 Ω 称为集合 C 的一个分类，$C_1,C_2,...,C_n$ 称为 C 的子种群。从中可以看出，C 本身是集合 C 的最大子种群。

我们对 Neighbor-joining 方法用计算机模拟，可以描述如下：

1. 采用 Kimura 双参数模型计算出任意两个序列之间进化距离 d_{ij}。
2. 计算第 i 个序列（即第 i 个叶子节点）的净分歧度 r_i，其计算公式为：

$$r_i=\sum_{k=1}^{n}d_{ik} \tag{4.7}$$

其中，n 表示叶子节点的个数，d_{ik} 表示第 i 个叶子节点和第 k 个叶子节点之间的进化距离。

3. 计算第 i 个叶子节点和第 j 个叶子节点之间的速率校正距离 M_{ij}，并筛选出最小速率校正距离。M_{ij} 的计算公式为：

$$M_{ij}=d_{ij}-\frac{r_i+r_j}{n-2} \tag{4.8}$$

4. 假设筛选出的最小速率校正距离存在在第 i 个叶子节点和第 j 个叶子节点之间，则定义一新节点 t，t 的左孩子节点是叶子节点 i，t 的右孩子节点是叶子节点 j，因此节点 t 与其他节点 k 的进化距离 d_{kt} 的计算公式为：

$$d_{kt}=\frac{d_{ik}+d_{jk}-d_{ij}}{2} \tag{4.9}$$

5. 从节点集合 L 中删除第 i 个节点 i 和第 j 个节点，并将节点 t 添加到节点集合中，叶子节点的个数 n 减去 1。

6. 判断 n 是否大于 2，如果大于 2 重复步骤 2-6，直到 n 的值为 2，则进化树完全构建。

4.2.4 利用模式匹配的序列相似性分析方法的基本步骤

1. 输入 N 条 DNA 序列 $S_1, S_2, ..., S_n$；
2. 使用基于模式匹配的多序列比对方法对这 N 条序列进行序列比对，生成结果序列集 $L = \{S_1', S_2', ..., S_n'\}$；
3. 对结果序列集 L 使用 Kimura 双参数模型计算进化距离矩阵；
4. 对进化距离矩阵计算并筛选最小速率校正距离，并找出最小速率校正距离所对应的结果序列 S_i' 和 S_j'；
5. 为系统进化树创建一新节点 t，t 的孩子节点为 S_i' 和 S_j'，从结果序列集 L 中删除节点 S_i' 和节点 S_j' 并加入新节点 t；
6. 重复 3-5，直到结果序列集 L 中节点个数为 0 时停止，则系统进化树构建完毕。

4.3 实验结果分析

使用第三章中进行序列比对的 DNA 序列数据进行实验。第一组是 6 种西藏部分地区豆科植物的根瘤菌 RDNA 序列，平均长度为 1419，如表 3.1 所示。第二组是人、山羊、山鸡、家属、老鼠、大猩猩六个物质的 β-球蛋白基因第一个外显子的 DNA 序列，平均长度为 91，如表 3.4 所示。第三组是 8 种 H5N1 型禽流感病毒的 HA 片断基因序列，平均长度为 1700，如表 3.7 所示。

对第一组数据采用基于模式匹配的多序列比对方法进行比对，得到 6 种根瘤菌的序列比对结果，采用 Kimura 双参数模型构建进化矩阵，如表 4.2 所示。将表 4.2 给出的进化矩阵输入到 PHYLIP 软件包中的 neighbor.exe 软件中，构造出物种的进化树，如图 4.1 所示。

表 4.2 6 种西藏根瘤菌 RDNA 全序列的进化矩阵

物种	Rhizobium mongolense	Rhizobium gallicum	Rhizobium yanglingense	Rhizobium leguminosarum	Sinorhizobium meliloti	Sinorhizobium morelense
Rhizobium mongolense	0.0000	0.0078	0.0044	0.0302	0.0461	0.0351
Rhizobium gallicum	0.0078	0.0000	0.0034	0.0313	0.0472	0.0355
Rhizobium yanglingense	0.0044	0.0034	0.0000	0.0278	0.0429	0.0319
Rhizobium leguminosarum	0.0302	0.0313	0.0278	0.0000	0.0426	0.0422
Sinorhizobium meliloti	0.0461	0.0472	0.0429	0.0426	0.0000	0.0173
Sinorhizobium morelense	0.0351	0.0355	0.0319	0.0422	0.0173	0.0000

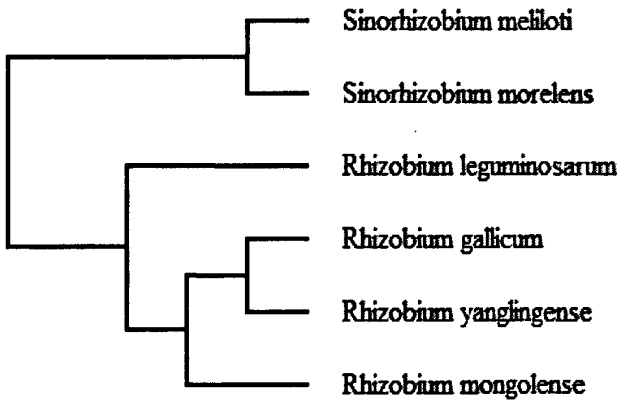

图 4.1 6 种西藏根瘤菌的进化树

观察表 4.2 和图 4.1 可知，6 种西藏根瘤菌的系统发育树基本分成 Sinorhizobium 和 Rhizobium 2 个分支。这与许多研究者的结果一致[76]，但 Rhizobium 分支中 gallicum 和 yanglingense 亲缘关系最近，而通过文献[60]的描述可知 Rhizobium mongolense 和 Rhizobium gallicum 的相似性才是最高，与事实不太符合。这是由于本算法更多是重视数学上的高匹配率，而忽略了序列比对本身的生物学意义，使得比对的结果虽然有高的 SPS 值，但并没有得到与事实相符的进化树。

对第二组数据采用基于模式匹配的多序列比对方法进行比对，得 11 种物种第一个外显子的序列比对结果，采用 Kimura 双参数模型构建进化矩阵，如表 4.3 所示。将表 4.3 给出的进化矩阵输入到 PHYLIP 软件包中的 neighbor.exe 软件中，构造出物种的进化树，如图 4.2 所示。

表 4.3 11 个物种第一个外显子的进化矩阵

物种	Human	Goat	Gallus	Rat	Mouse	Chimpanzee	Bovine	Gorilla	Opossum	Lemur	Rabbit
Human	0.0000	0.1385	0.1385	0.2280	0.2747	0.0142	0.1055	0.0142	0.2858	0.3611	0.1332
Goat	0.1385	0.0000	0.0000	0.2608	0.2711	0.1218	0.0384	0.1218	0.2916	0.3568	0.1675
Gallus	0.1385	0.0000	0.0000	0.2608	0.2711	0.1218	0.0384	0.1218	0.2916	0.3568	0.1675
Rat	0.2280	0.2608	0.2608	0.0000	0.1742	0.2092	0.2608	0.2092	0.4174	0.4930	0.2686
Mouse	0.2747	0.2711	0.2711	0.1742	0.0000	0.2686	0.3127	0.2686	0.5210	0.5655	0.3672
Chimpanzee	0.0142	0.1218	0.1218	0.2092	0.2686	0.0000	0.0895	0.0000	0.2916	0.3552	0.1166
Bovine	0.1055	0.0384	0.0384	0.2608	0.3127	0.0895	0.0000	0.0895	0.2654	0.3127	0.1332
Gorilla	0.0142	0.1218	0.1218	0.2092	0.2686	0.0000	0.0895	0.0000	0.2916	0.3552	0.1166
Opossum	0.2858	0.2916	0.2916	0.4174	0.5210	0.2916	0.2654	0.2916	0.0000	0.5967	0.3568
Lemur	0.3611	0.3568	0.3568	0.4930	0.5655	0.3552	0.3127	0.3552	0.5967	0.0000	0.3784
Rabbit	0.1332	0.1675	0.1675	0.2686	0.3672	0.1166	0.1332	0.1166	0.3568	0.3784	0.0000

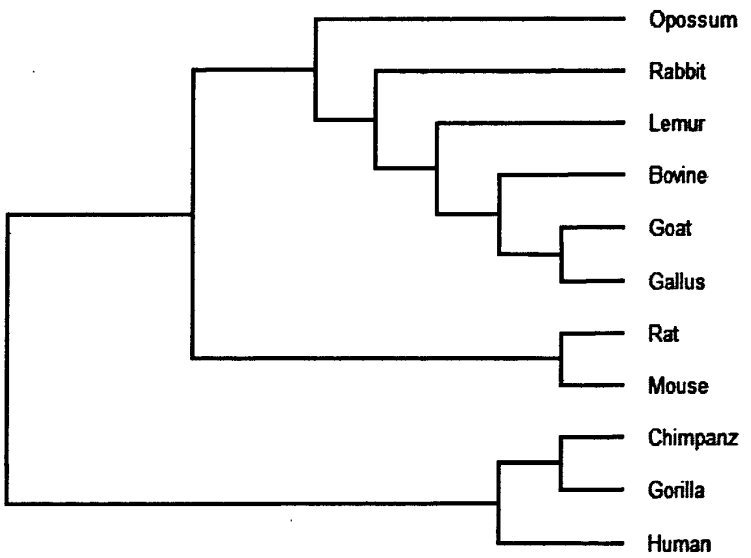

图 4.2 11 种物种的进化树

观察图 4.2 可知，Chimpanzee（黑猩猩）和 Gorilla（大猩猩）最相似，相对于其他物种，Human（人类）与 Chimpanzee（黑猩猩）和 Gorilla（大猩猩）最为相似，比较符合实际已有的结果。Rat（小鼠）和 Mouse（老鼠）同为鼠类，其 DNA 序列应有较大的相似性，表 4.3 和图 4.2 都可以看出比较相似。第二组实验也出现和第三组实验同样的情况，由于序列比对的结果只重视了高匹配率而忽略了生物学意义，使得 Gallus（山鸡）虽然为非哺乳动物，其余十种为哺乳动物，表 4.2 和图 4.2 并没有把 Gallus（山鸡）和

其余十种动物区分开来。

对第三组数据采用基于模式匹配的多序列比对方法进行比对，得到 8 种 H5N1 型禽流感病毒的 HA 片断基因的序列比对结果，采用 Kimura 双参数模型构建进化矩阵，如表 4.4 所示。将表 4.4 给出的进化矩阵输入到 PHYLIP 软件包中的 neighbor.exe 软件中，构造出物种的进化树，如图 4.3 所示。

表 4.4 8 种 H5N1 型禽流感病毒的 HA 片断基因的进化矩阵

物种	DG12	DG40	DG1681	DG1793	DH1265	DH1608	CY447	CH18
DG12	0.0000	0.0172	0.0513	0.0516	0.0482	0.0495	0.0526	0.0475
DG40	0.0172	0.0000	0.0356	0.0358	0.0334	0.0347	0.0378	0.0321
DG1681	0.0513	0.0356	0.0000	0.0026	0.0520	0.0533	0.0543	0.0507
DG1793	0.0516	0.0358	0.0026	0.0000	0.0523	0.0536	0.0546	0.0510
DH1265	0.0482	0.0334	0.0520	0.0523	0.0000	0.0047	0.0334	0.0148
DH1608	0.0495	0.0347	0.0533	0.0536	0.0047	0.0000	0.0359	0.0160
CY447	0.0526	0.0378	0.0543	0.0546	0.0334	0.0359	0.0000	0.0340
CH18	0.0475	0.0321	0.0507	0.0510	0.0148	0.0160	0.0340	0.0000

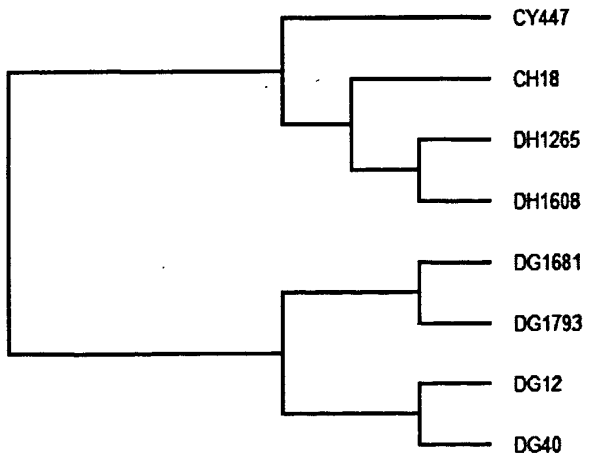

图 4.3 8 种 H5N1 型禽流感病毒的 HA 片断基因的进化树

由图 4.3 可知，在同种动物身上采集的病毒中相同年份和相同地域的相似性最高。另外，使用 CLUSTALX 软件进行多序列比对，同样采用 Kimura 双参数模型和 neighbor.exe 程序构建进化树，这两种方法得到的进化树的结构是完全相同的。并且文献[26]使用基于 BB 信息离散度的 DNA 序列相似性分析的方法也得到了相同的结果。

4.4 小结

本章首先介绍了四种常用的序列相似性分析的方法，并着重介绍了利用模式匹配的序列相似性分析方法的基本步骤。因为 Kimura 两参数模型更能体现生物的突变规律，计算出的序列距离更为精确，所以我们采用基于模式匹配的多序列比对，使用 Kimura 双参数模型和 Neighbor-joining 方法实现相似性分析。主要步骤是首先采用基于模式匹配的 DNA 多序列比对算法得到比对结果，采用 Kimura 双参数模型计算进化矩阵，使用 PHYLIP 软件包中的 neighbor.exe 构造物种进化树。实验结果表明，该方法能够对 DNA 序列的相似性进行有效分析，分析结果接近事实。

结 论

人类一直在探寻着生命起源的奥秘，追寻着生物进化的初衷。随着生命科学研究进入到后基因时代之后，已经收集了海量的生物信息数据存储在各种数据库中。如何在这庞大的数据信息中抽取我们有用的信息一直是困扰我们的一个难题。由此，一门新兴学科，也是一门涉及多学科的交叉学科，生物信息学诞生了。生物信息学成为生命科学研究的重要组成和前沿领域。

生物信息学中最基本和最核心的问题之一就是多序列比对。由于多序列比对处理的数据越来越庞大和复杂，所以其算法对计算精度和运算速度的要求也越来越高。如何能快速有效的获得比对的结果，一直苦恼着众多的学者们。另外序列相似性分析方法中最常用就是序列比对的方法，怎样使得序列相似性分析的结果更接近真实情况。针对这两个问题，本文做了如下工作：

1. 系统介绍了生物信息学的相关知识，多序列比对、序列相似性分析和构建进化树的研究现状，并重点介绍了经典的多序列比对算法和进化树的构建方法。

2. 针对多序列比对这个NP难问题，许多学者已经提出了不同的解决方法。本文提出了一种基于模式匹配的新方法，以DNA多序列全局比对的星比对算法为基础，将模式匹配应用于其中。并对基于关键字树的星比对算法进行改进，在寻找中心序列时首先建立一颗公共的模式树，通过每个序列的模式号的比较逐级去掉希望最小的序列，在降低多序列比对的时间复杂度的同时提高了比对的准确性，使比对的结果更有助于生物实验。

3. 本文对提出的基于模式匹配的多序列比对算法进行了实验分析。实验结果表明，原星比对算法和基于关键字树的星比对算法的比较，验证了该算法的有效性。

4. 将本文提出的基于模式匹配的多序列比对方法应用到序列相似性分析中，对比对结果使用Kimura两参数模型和Neighbor-joining方法构建进化树，也通过实验验证了使用该方法构建的进化树和实际情况基本相符。

多序列比对算法仍然是生物信息学中的一个研究难题，如何使用更加快速有效的方法来构建进化树也是许多学者的研究目标。由于实验环境的限制和本人能力问题，本文的研究工作还需要更加完善的改进，主要体现在：

1. 寻找中心序列仍然存在着不安全因素，由于此方法将每个序列截断成相同的子串，只考虑了子串的出现频率而忽略了少于子串长度的子序列的出现频率，势必会中心序列的确定。

2. 本文只从程序平均运行时间和 SPS 分值对该算法进行评价，在实际的多序列比对过程中，还需要考虑序列的条数，序列的长度，CS 分值等。因此，应该从更多的方面去评价一个算法的有效性。

3. 本文使用比对结果构造进化树时，构造的进化树不是完全与事实相符。所以，本算法还有待改进。

参考文献

[1] 孙啸, 陆祖宏, 谢建明. 生物信息学基础. 北京: 清华大学出版社, 2005, 10-17

[2] Carrillo H, Lipman D J. The multiple sequence alignment Problem in biology. SIAM Journal of Applied Mathematics, 1988, 48(5): 1073-1082

[3] 唐玉荣, 一种优化的生物序列比对算法. 计算机工程与设计, 2004, 25(11): 1936-1945

[4] Hogeweg P, Hesper B. The alignment of sets of sequences and the construction of phyletic trees: an integrated method. J Mol Evol, 1984, 20(2): 175-186

[5] Feng D F, Doolittle R F. Progressive sequence alignment as a prerequisite to correct phylogenetic trees. J Mol Evol, 1987, 25(4): 351-360

[6] Taylor W R. A flexible method to align large numbers of biological sequences. J Mol Evol, 1988, 28: 161-169

[7] 段敏, 许龙飞. 生物 DNA 序列比对算法研究. 佳木斯大学学报, 2005, 23(2): 153-158

[8] 向昌盛, 周建军, 周子英. 模拟退火遗传算法在生物多序列比对中的应用. 湖南农业科学, 2008, (4): 29-34

[9] 徐小俊, 雷秀娟, 郭玲. 基于 SWGPSO 算法的多序列比对. 计算机工程, 2011, 37(6): 184-186

[10] Stoye J, Moulton V, Dress A W. DCA: an efficient implementation of the divide and conquer approach to simultaneous multiple sequence alignment. Comput Appl Biosci, 1997, 13(6): 625-626

[11] Reinert K, Stoye J, Will T. An iterative method for faster sum-of-pair multiple sequence alignment. Bioinformatics, 2000, 16(9): 808-814

[12] 业宁, 张倩倩, 许翠云. 一种多序列比对分值算法 DCA-ClustalW. 计算机与数学工程, 2010, 38(11): 30-33

[13] 陈娟, 陈峻. 多重序列比对的蚁群算法. 计算机应用, 2006, 26(6): 124-128

[14] 彭东海, 骆嘉伟, 袁辉勇. 基于改进蚁群算法的多序列比对. 计算机工程与应用, 2009, 45(33): 114-119

[15] 李方洁, 刘希玉. 基于渐进蚁群算法的 DNA 多序列比对. 网络安全技术与应用, 2010, (9): 78-80

[16]Gibbs A J, McIntyre G A. The diagram a method for comparing sequences its use with amino and nucleotide sequences. Eur J Biochem, 1970, 16: 1-11

[17]Needleman S B, Wunsch C D. A general method applicable to the search for similarities in the amino acid sequences of two proteins. Journal of Molecular Biology, 1970, 48: 443-453

[18]Smith T F, Waterman M S. Identification of common molecular subsequences. J Mol Biol, 1981, 147: 195-197

[19]Randic M. Graphical representations of DNA as 2-D map. Chemical Physics Letters, 2004, 386(2): 468-471

[20]Li C, Wang J. New Invariant of DNA Sequences. Journal of Chemistry Information and Modeling, 2005, (45): 115-120

[21]Zhang Y S, Wei C. Invariant of DNA sequences based on 2DD-curves. Journal of Theoretical Biology, 2006, 242: 382-388

[22]张惜珍. DNA序列3D图形表示及进化树算法研究: [湖南大学硕士学位论文]. 长沙: 湖南大学, 2007, 39-40

[23]汪挺松. 曲率在生物序列相似性分析中的应用: [大连理工大学硕士学位论文]. 大连: 大连理工大学, 2007, 42-45

[24]李梅, 白凤兰. 基于 DTW 距离的 DNA 序列相似性分析. 生物数学学报, 2009, 24(2): 374-378

[25]唐晓婵. 基于 4D 图形表示的 DNA 序列相似性分析. 科学通报, 2010, (6): 442-446

[26]刘芳. 基于信息离散度的 DNA 序列相似性分析研究: [湖南大学硕士学位论文]. 湖南: 湖南大学, 2009, 33-40

[27]Morrison D A. Phylogenetic tree-building. International Journal for Parasitology, 1996, 26(6): 589-617

[28]Fitch W M, Margoliash E. Construction of phylogenetic trees. Science, 1967, 155: 279-284

[29]Saitou N, Nei M. The neighbor-joining method: a new method for reconstructing phylogenetic trees. Molecular Biology and Evolution, 1987, 4(4): 406-425

[30]Felsenstein J. Evolutionary trees from DNA sequences: a maximum likelihood approach. Journal of Moleculer Evolution, 1981, 17(6): 368-376

[31]Carnin J H, Sokal R R. A method for deducing branching sequences in phylogeny.

Society for the Study of Evolution, 1965, 19(3): 311-326

[32]Ota S, Li W H. NJML: a hybrid algorithm for the neighbor-joining and maximum-likelihood methods. Moleeular Biology and Evolution, 2000, 17(9): 1401-1409

[33]Vincent R, Olivier G. Improvement of distance based phylogenetic methods by a local maximum likelihood approach using triplets. Molecular Biology and Evolution, 2002, 19(11): 1952-1963

[34]谭严芳, 金人超. 一种基于 NJ 的高效构建系统进化树算法. 计算机工程与应用, 2004, 21: 84-86

[35]徐立业, 李玉鑑. UPGMA 树的不惟一性问题及其解决方法. 生物信息学, 2007, 5(4): 67-71

[36]郑文新. 冠状病毒进化关系分析: [天津大学硕士学位论文]. 天津: 天津大学, 2005, 36-40

[37]Liao B, Zhu W, Liu Y. 3D graphical representation of DNA sequence without degeneracy and its application in constructing phylogenetic tree. MATCH Communications in Mathematical and in Computer Chemistry, 2006, 56: 209-216

[38]张建辉. 基于 Z 曲线理论的冠状病毒进化关系分析: [天津大学硕士学位论文]. 天津: 天津大学, 2007, 42-45

[39]柳菁筠, 李大超. 用基于模糊聚类的 Kruskal 算法构建进化树. 海南师范大学学报(自然科学版), 2007, 20(4): 303-306

[40]苏志忠. 聚类分析研究及其在生物数据分析中的应用: [湖南大学硕士学位论文]. 长沙: 湖南大学, 2007, 37-42

[41]李刚成, 刘赞波, 曾庆光. 一种基于模糊聚类的构造进化树方法. 计算机应用, 2009, 29(3): 836-839

[42]邹权, 郭茂祖等. 基于关键字树的 DNA 多序列星比对算法. 电子学报, 2009, 37(8): 1746-1750

[43]张春霆. 生物信息学的现状与展望. 世界科技研究与发展, 2000, 22(6): 17-20

[44]魏静. 并行遗传算法在生物序列比对中的应用研究: [天津大学硕士学位论文]. 天津: 天津大学信息工程学院, 2004, 11-12

[45]Kanehisa M. Post-Genome Informatics. Oxford: Oxford University Press, 2000, 6-9

[46]Attwood T K, Parry-Smith D J. 生物信息学概论. 罗静初译. 北京: 北京大学出版社, 1999, 168-196

[47]肖智伟. 基于最大权值路径算法的DNA多序列比对方法: [西安电子科技大学硕士学位论文]. 西安: 西安电子科技大学, 2006, 8-9

[48]Gusfield D. Algorithms on Strings, Trees and Sequences: Computer Science and Computation Biology. Cambridge University Press, 1997, 332-366

[49]Altschul S F. Gap costs for multiple sequence alignment. J Theor Biol, 1989, 138: 297-309

[50]Murata M, Richardson J S, Sussman J L. Simultaneous comparison of three protein sequences. Proc Natl Acad Sci, 1985, 82: 3073-3077

[51]Brown J W. The Ribonuclease P database. Nucleic Acids Research, 1999, 27(1): 314

[52]Masatoshi N, Sudhir K. 分子进化与系统发育. 吕宝忠译. 北京: 高等教育出版社, 2002, 65-67

[53]高凯. NJ进化树构建方法的改进及其应用: [北京工业大学硕士学位论文]. 北京: 北京工业大学, 2008, 9-10

[54]Navarro G, Raffinot M. Flexible Pattern Matching in Strings: Practical on-line search algorithms for texts and biological sequences. New York: Cambridge University Press, 2002, 36-128

[55]卢开澄. 计算机算法导引-设计与分析. 北京: 清华大学出版社, 1996, 221-226

[56]Knuth D E, Morris J H, Pratt V R. Fast pattern matching in strings. SIAM Journal on Computing, 1977, 6(2): 323-350

[57]Boyer R S, Moore J S. A fast string searching algorithm. Communications of the ACM, 1977, 20(10): 762-772

[58]Aho A V, Corasick M J. Efficient string matching: an aid to bibliographic search. Communications of the ACM, 1975, 18(6): 333-340

[59]Shin S Y, Lee I H, Kim D, et al. Multiobjective Evolutionary Optimization of DNA Sequences for Reliable DNA Computing. IEEE Transactions on Evolutionary Computation, 2005, 9(2): 143-158

[60]王素英, 杨晓丽等. 西藏根瘤菌新表现群的DNA同源性及16S rDNA全序列分析. 微生物学报, 2006, 46(1): 132-135

[61]Glatin L. Information Theory and The Living System. New York: Columbia University Press, 1978, 21-48

[62]Fang W W, Roberts F S, Ma Z R. A Measure of Discrepancy of Multiple Sequences. Information Science, 2001, 137: 75-102

[63]孙啸, 傅静, 焦典等. 利用序列统计特征分析基因组序列. 见: 生物信息学若干前沿问题的探讨 中国科协第 81 次青年科学家论坛论文集. 合肥: 中国科学技术大学出版社, 2004, 51-56

[64]Karlin S, Ladunga I. Comparisons of eukaryotic genomic sequences. Proc Natl Acad Sci, 1994, 91: 12832-12836

[65]姚玉华. 生物序列相似性分析的图形表示及其不变量方法: [大连理工大学博士学位论文]. 大连: 大连理工大学, 2006, 20-35

[66]Zhang C T, Zhang R, Ou H Y. The Z curve database: a graphic representation of genome sequences. Bioinformatics, 2003, 19(5): 593-599

[67]Gates M A. A simple way to look at DNA. Journal of Theoretical Biology, 1986, 119: 319-328

[68]Nandy A. A new graphical representation and analysis of DNA sequence strueture I. Methodology and application to globin genes. Current Science, 1994, 66: 309-313

[69]Leong P M, Morgenthaler S. Random walk and gap plots of DNA sequences. Computer Application Bioscience, 1995, 11(5): 503-550

[70]Liao B. A 2D-Graphical Representation of DNA Sequence. Chemical Physics Letters, 2005, 401(12): 196-199

[71]Zhang Y S, Tan M S. Visualization of DNA Sequences Based on 3DD-Curves. Journal of Mathematical Chemistry, 2008, 44: 206-216

[72]Qi Z H, Qi X Q. Novel 2D Graphical Representation of DNA Sequecne Based on Dual Nucleotides. Chemical Physics Letters, 2007, 440: 139-144

[73]Cao Z, Liao B, Li R F. A Group of 3D Graphical Representation of DNA Sequences Based on Dual Nucleotides. International Journal of Quantum Chemistry, 2008, 108(9): 1485-1490

[74]孙啸, 陆祖宏, 谢建明. 生物信息学概论. 清华大学出版社, 2004: 92-96

[75]Maeda N, Bliska J. Molecular evolution of intergenic DNA in higher primates: pattern of DNA changes, molecular clock and evolution of repetitive sequences. Molecular Biology Evolution, 1988, 5(7): 1-20

[76]韦革宏, 朱铭莪, 陈文新. 鸡眼草根瘤菌的 16S rDNA 全序列分析. 微生物学报, 2001, 41(1): 113-116

附录 A（攻读硕士期间参加的项目）

[1] 国家自然科学基金[60873184]：新型表达模式下的功能基因分析算法研究

[2] 湖南省财政厅项目[湘财教字[2010]163 号]：蛋白质组信息获取和分析方法研究

致 谢

在此，首先要向我的导师骆嘉伟教授致以最诚挚的谢意。骆老师一丝不苟的工作态度、严谨的科研作风、追求科学真理的精神使我深受启迪，给我以后的学习和生活树立了良好的榜样。同时，这篇论文的完成也耗费了骆老师不少的精力，论文的选题、资料的搜集、疑难问题的解答、论文的修改，每个阶段都倾注了她不少的心血。我的每一点点的进步都包含着骆老师严格的要求和悉心的指导。所以，我再次对骆老师表示深深的谢意！

另外，感谢我的同学徐畅、宋颖、肖坚、文星、余宇华、杨丽，与他们的讨论总让我受益匪浅。

同时感谢我的同事李锡辉、黄海芳、赵莉、方丽，在学习和生活中的帮助，有了他们，才能让我的论文更顺利地完成。

特别感谢我的父母。他们无时无刻都在给予我最大的关怀和爱护，有了他们的支持，才会有今天的成功。

最后，在此感谢所有帮助和支持过我的朋友，谢谢你们。

基于模式匹配的DNA多序列比对及相似性分析

作者： 王樱
学位授予单位： 湖南大学