ТЕХНИЧЕСКИ УНИВЕРСИТЕТ - СОФИЯ

Факултет по приложна математика и информатика

КУРСОВ ПРОЕКТ

по

Комбинаторни алгоритми и приложения

на тема

Разпознаване на изображения с невронни мрежи

Изготвили:

Мария Живкова Желязкова фак. № 991319004

Илия Георгиев Къркъмов фак. № 991319006

Георги Стоянов Арътлъков фак. № 991319005

Увод

В рамките на текущия курсов проект е разгледан подход за решаване на задачата за множествено подравняване на секвенции (MSA). Избраният подход за решаване на задачата представлява евристичен модел за оптимизация по метода на мравчените колонии. Този подход за решаване на проблема позволява лесна паралелизация на задачата, така че да бъдат постигнати много добри резултати по отношение на ефективност и производителност.

Основната задача е реализирането на приложение, което да подравнява произволен брой подадени секвенции. За целта е използван езикът за програмиране JAVA. Важна част от разработката на приложението включва, ефективно използване на процесорната мощ на машината върху, която се изпълнява приложението.

Множествено подравняване на секвенции

Множествено подравняване на секвенции е един от най-важните и трудни задачи в полето на молекулярната биология и биоинформатиката. Едновременното изравняване на много нуклеотидни или аминокиселинни секвенции е основен инструмент в молекулярната биология и биоинформатика и играе ключова роля за откриване на региони със значително сходство на секвенции от колекции от първични секвенции на нуклеинова киселина или протеини. Множественото подравняване може да се използва и за подпомагане на реконструкцията на филогенетични дървета, за намиране на модели за характеризиране на белтъчни семейства, за откриване на хомология между нови секвенции и съществуващи и прогнозиране на вторичната и третичната структура на протеиновите последователности.

Основата на подравняването на секвенции и множественото им подравняване е еволюционната теория. Според нея по време на еволюцията се случват мутации в организмите и така се създават различия между семейства от видове. Повечето от тези промени се дължат на локални мутации. Локалните мутации се изразяват в три различни операции - добавяне, изтриване и заместване. Добавянето вмъква определена поредност от букви в секвенцията, изтриването премахва поредица от секвенцията и заместването замества дадена поредица от секвенцията с друга поредица от букви.

При сравнение на две секвенции се откриват позициите в двете секвенции, за които се счита, че са функционално и еволюционно свързани. Когато се разглежда семейство от секвенции, целта е да се открият сходствата в цялото семейство. В този случай не може да се приложи сравняване на всички секвенции две по две и затова се прибягва към множественото подравняване. В множественото подравняване се разкриват скрити мотиви като те се представят като колони с много по-малка вариация в сравнение с тези около тях.

Множественото подравняване е формулиран като задача за оптимизация, когато се използва обективна функция от биологични принципи. Това е задача с нелинейната оптимизация, дефинирана в дискретно пространство, чието изчислително време е много дълго. За целта се използват приблизителни методи за оптимизация, които гарантират намирането на приблизително оптимално решение в кратко време.

Оптимизация с мравчени колонии

Оптимизационният алгоритъм с мравчени колонии е вероятностен подход за решаване на изчислителни задачи, който може да бъде сведен до откриване на добри пътища през граф. За пръв път, алгоритъма е предложен от проф. Марко Дориго през 1992 година, като от тогава той претърпява много изменения с цел да се решават повече и по-комплексни проблеми.

В реалността, истинските мравки (първоначално) се движат по случаен начин и при откриване на източник на храна се завръщат към гнездото на колонията си, като по целия обратен път полагат следи от вещество, наречено феромон, което служи за комуникация с останалите мравки. Ако други мравки се натъкнат на маршрута на първата мравка, те с голяма степен на вероятност престават да се движат по случаен начин, а поемат по оставената следа от феромон до източника на храна и в случай, че също открият храна, при връщането си полагат собствена следа от феромон върху вече съществуващата. По този начин количеството феромон се увеличава и маршрута до източника на храна става още по-привлекателен за следващите мравки.

Феромонът има свойството да изветрява с времето, като по този начин даденият път става не атрактивен за мравките. По този начин те успяват да елиминират по-дългите пътища и започват да използват само по-кратките.

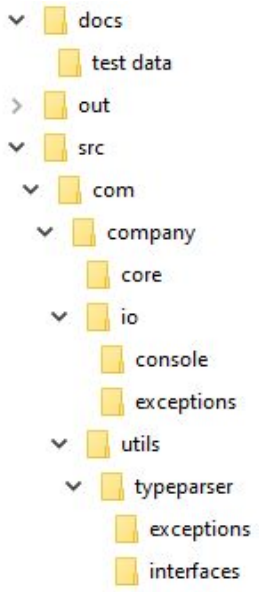
В контекста на алгоритъма на мравчените колонии, изветряването на феромона представлява предимство, тъй като се избягва сходимостта на алгоритъма към решение, представляващо локален оптимум. Ако не съществуваше ефектът за изпарението, то пътищата избрани от първите "мравки" биха имали тенденцията да са прекомерно привлекателни за всички следващи, което на свой ред би ограничило значително изследваната от "мравките" област на решенията.

По този начин, когато една мравка открие добър (т.е. кратък) път от мравуняка си до някакъв източник на храна, с голяма вероятност и други мравки ще последват пътя й и постепенно положителната обратна връзка ще доведе до това всички мравки да следват един и същ път. Идеята на алгоритъма за оптимизация по метода на мравчената колония е да се имитира това поведение с "изкуствени мравки", които се движат по дъгите на граф, представляващ областта на възможните решения, като откритият от мравките път представлява оптималното от тези решения.

Имплементация

Файлова структура

За имплементация на множествено подравняване е реализиран проект на програмния език JAVA версия 13. На фиг. 1 е показана структурата на проекта.

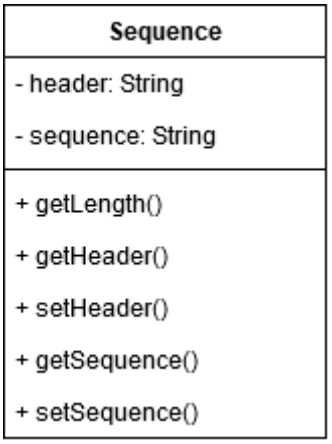


Фиг. 1 - Файлове структура на проекта

Файловата структура е типична за проекта реализиран на Java. Папката ***docs*** съдържа документация на курсовия проект и тестови данни, които се намират в подпапката ***test data***. Папката ***out*** е създадена от средата за разработка, която е IntelliJ на Jetbrains. В нея са съхранени компилираните класове. Папката ***src*** съдържа програмният код, като всяка подпапка в нея е пакет в Java.

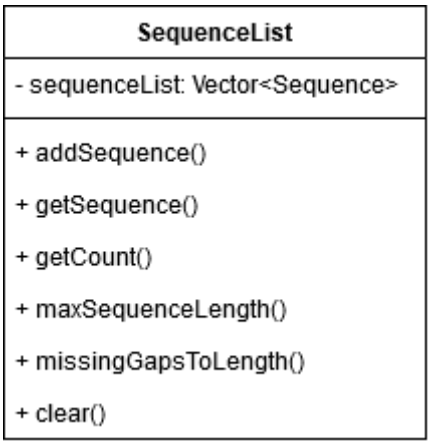
Реализирани класове

Класът Sequence съдържа методи за управлението на една секвенция.



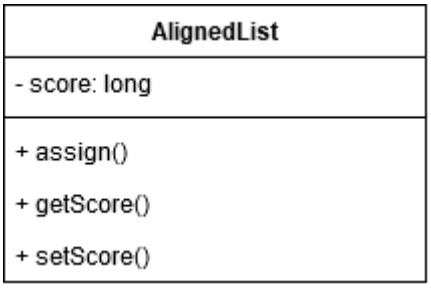
Фиг. 2 - Диаграма на класа Sequence

Класът SequenceList съдържа списък от секвенции и е отговорен за достъпването, добавянето и премахването им. Също така имплементира методи за откриването на всички празни места в тях, както и техните дължини.



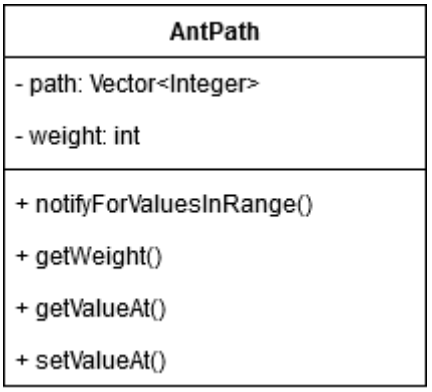
Фиг 3 - Диаграма на класа SequenceList

Класът AlignedList е наследник на класа SequenceList. Неговата основна цел е да съдържа вече подредените секвенции и тяхната оценка.



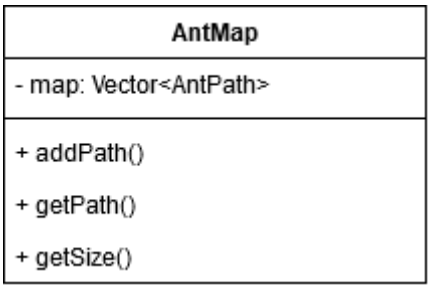
Фиг 4 - Диаграма на класа AlignedList

Класът AntPath задава път изминат от мравката като също така се пази и неговото тегло. Класът предоставя методи за достъпване и изменяне на параметрите в пътя.



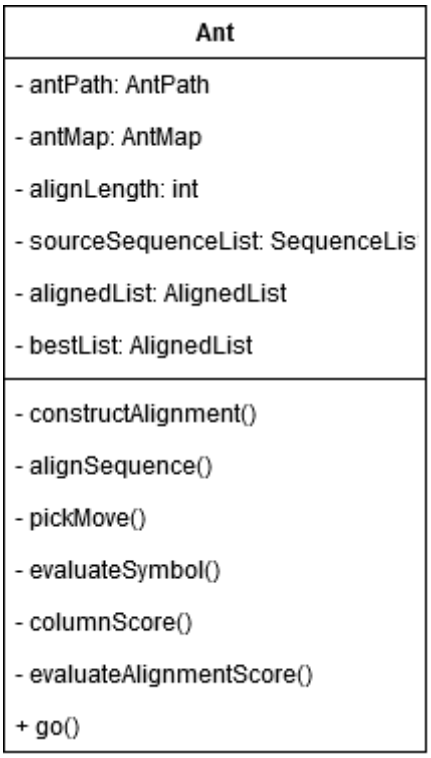
Фиг 5 - Диаграма на класа AntPath

Класът AntMap съдържа списък с всички възможни пътища, изминати от мравки. Предоставя възможността за добавяне и достъпване на пътища. Основната цел на този клас е да съдържа всички най-добри пътища изминати от предходни мравки, като по този начин се помага на мравките да открият по-късия път измежду всички останали.



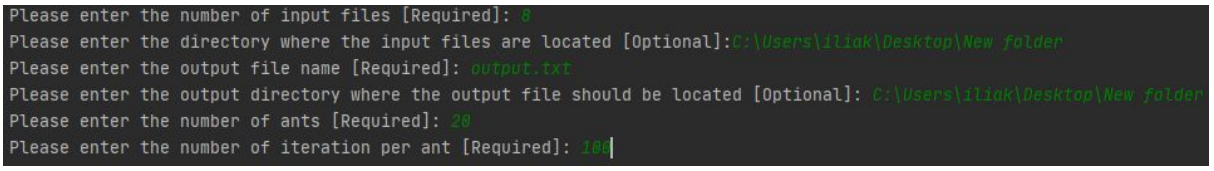
Фиг 6 - Диаграма на класа AntMap

Класът Ant съдържа основната логика за подравняването на семейство от секвенции. В този клас се генерира оценката на полученото подравняване на семейството от секвенции. Оценъчната функция дава 10 точки ако има съвпадение, отнема 2 точки ако няма съвпадение и ако има празно място отнема 5 точки. Основният метод на този клас е go(). На него като параметър се подава колко итерации трябва да направи една мравка преди да приключи, а той от своя страна създава и оценява подредбите и при наличието на по-добър път го добавя е инстанцията на класа AntMap.



Фиг 7 - Диаграма на класа Ant

При стартиране на приложението трябва да бъдат въведени основни параметри за работа. Основните са броят на файловете (всеки файл отговаря на една секвенция във формат FASTA), броят на мравките и броят итерации за всяка мравка. Фиг. 8 показва процесът по въвеждане на параметрите.



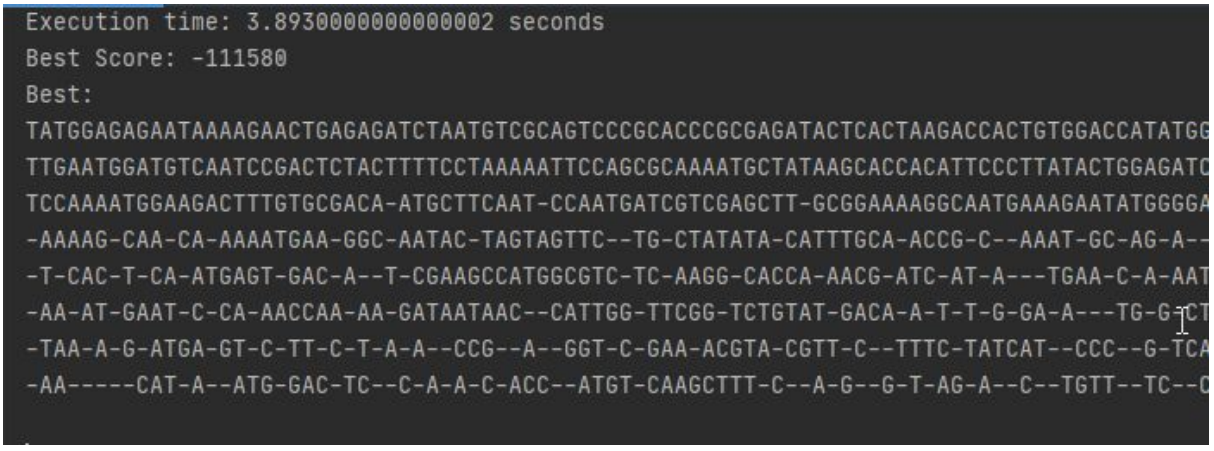
Фиг. 8 - Процес по въвеждане на параметри за работа

Приложението очаква файловете съдържащи секвенциите да бъдат именувани с имена от 1 до броят на секвенциите. Например за параметрите въведени в фиг. 8, броят на файловете е 8 следователно приложението ще се опита да прочете файлове 1.txt, 2.txt, …, 8.txt.

При стартиране на приложението то създава списък от нишки (броят им е равен на броя логически нишки по две). На всяка нишка ще работи по една мравка, като ако броят на мравките е по-голям от броя на нишките, то те влизат в опашка и изчакват някоя нишка да приключи работа. Всички мравки споделят обекта AntMap, в който добавят добрите намери пътища. По този начин се осъществява комуникация между отделните мравки и се подпомага намирането на оптималното решение.

Проведени тестове

За тестването на алгоритъма е използвано семейството от секвенции за Influenza вируса, което съдържа осем сегмента от секвенции. На фиг. 9 е показана извадка на резултата от изпълнението на приложението.



Фиг. 9 - Резултат от изпълнение на приложението

Извадката от фиг. 9 е изпълнена със 20 мравки, като всяка мравка извършва по 20 на брой итерации. На извадката се вижда част от подредбата на секвенциите. Крайната оценка е отрицателна, защото секвенциите са с доста различна големина съответно са добавени много гапове, а това води до намаляване на оценката.