**МЕЖДУНАРОДНЫЙ УНИВЕРСИТЕТ ПРИРОДЫ, ОБЩЕСТВА И ЧЕЛОВЕКА «ДУБНА»**

Кафедра «Системного анализа и управления»

**Курсовая работа**

«Эволюционное моделирование – особенности, значение, приложения»

по дисциплине:

«Теория систем и системный анализ»

**Выполнил:**

студент гр. 4011П-2Д

Записочный В.В.

**Руководитель:**

к.т.н.

доцент кафедры САУ

Тятюшкина О.Ю.

ДУБНА

2013

Содержание

[Введение 4](#_Toc350549138)

[Глава 1. Основные принципы эволюционного моделирования. 6](#_Toc350549139)

[Глава 2. Теория и основы генетических алгоритмов. Стандартные модели ГА. Основные понятия и принципы. 11](#_Toc350549140)

[Глава 3. Обзор программных продуктов для генетических вычислений. 22](#_Toc350549141)

[Глава 4. Реализация и применение генетических алгоритмов 28](#_Toc350549142)

[Заключение 34](#_Toc350549143)

[Список используемой литературы 35](#_Toc350549144)

# Введение

Процессы эволюционных изменений, происходящие в естественных системах, издавна привлекают исследователей. Применение методов моделирования для изучения удивительной способности биологических систем к адаптации, приспособлению к условиям окружающей среды перспективно с точки зрения применения подобных принципов в искусственных технических системах, созданных человеком. Новое направление в области искусственного интеллекта, названное эволюционным моделированием (ЭМ), отвечеает за разработку и совершенствование теоретической базы, методов моделирования эволюционных изменений с применением средств математического и компьютерного моделирования.

Эволюционное моделирование определяют как «направление в искусственном интеллекте, в основе которого лежат принципы и понятийный аппарат, заимствованные из популяционной генетики и объединяющее компьютерные методы (генетические алгоритмы, генетическое программирование, эволюционное программирование и эволюционные стратегии) моделирования естественных эволюционных процессов» [1]. Возможно выделить аналогию между эволюционными процессами, происходящими в естественных системах, и задачами математического программирования. Схожесть указанных процессов заключается в том, что и эволюционные вычисления, и задачи математической оптимизации направлены на отыскание экстремума определенной величины. При этом накладываются дополнительные ограничения на значения переменных, что связано с ограниченностью доступных ресурсов. В случае естественных систем критерием оптимальности является приспособленность особи к условиям окружающей среды. В терминах математической модели оценку приспособленности возможно интерпретировать в виде оценки качества полученного решения относительно других исследованных точек пространства поиска. Существует и некоторое отличие – при решении математических задач оптимизации обычно ставят целью отыскание абсолютного экстремума, т.е. ведут поиск решения, оптимального на всем прострастве поиска. Заимствование терминологии, принципов и ключевых свойств эволюции – наследственности, изменчивости и естественного отбора – характерная черта, которая выделяет данную область в качестве самостоятельного направления исследований. Связь между задачами оптимизации и методами эволюционного моделирования дает основания для внедрения эволюционных стратегий на практике. Особенно интересным с данной точки зрения является применимость принципов эволюционных вычислений для решения задач, которые невозможно решить точными математическими методами. Трудности при решении подобных задач возникают из-за неполноты исходных данных, присутствия дополнительных ограничений, быстрого роста вычислительных затрат при повышении размерности,из-за динамично изменяющихся внешних условий и т.д.

В качестве "первопроходцев" в исследованиях механизмов эволюционного моделирования как нового независимого направления в области искусственного интеллекта, согласно [1], принято считать Джона Генри Холланда, Лоуренса Дж. Фогеля, А. Овена, М. Уолша, И. Букатову, Л. Расстригина. В основном содержание их работ связано с попытками воспроизвести в терминах математических моделей и компьютерных алгоритмов механизмы эволюционных измененеий, наблюдаемые в природных системах. В роли качестве доказательста рациональности применения таких алгоритмов исследователи выделяют постоянное стремление природы к достижению оптимальных форм существования при условии минимизации затрат ресурсов. В итоге в ряде научных работ была сформулирована теория эволюционного моделирования (или эволюционных вычислений). Затем опубликованные теоретические выкладки были использованы в качестве основы для разработки принципиально новых методов решения задач оптимизации.

# Глава 1. Основные принципы эволюционного моделирования.

## Эволюционное моделирование. Определение, назначение и области применения

Под эволюционным моделированием понимают «направление в искусственном интеллекте, в основе которого лежат принципы и понятийный аппарат, заимствованные из эволюционной биологии и популяционной генетики и объединяющие компьютерные методы (генетические алгоритмы, генетическое программирование, эволюционное программирование и эволюционные стратегии) моделирования эволюционных процессов в искусственных системах» [1].

Применение алгоритмов ЭМ в задачах оптимизации имеет несколько специфических особенностей. Данные особенности отражают разницу методов ЭМ и классических алгоритмов при решении задач оптимизации.

В качестве отличительных особенностей ЭМ выделим следующие:

* Применение кодирования данных в виде хромосомы, чаще всего в виде набора символов некоторого алфавита.
* В отличие от классических алгоритмов решения задач оптимизации, в эволюционных вычислениях поиск в пространстве решений производится не из единой точки, а из множества точек.
* В ЭМ есть возможность накапливать информацию в процессе исследования пространства решений. Накопленная при этом информация затем может быть использована для поиска еще более "приспособленных" решений. Следовательно, эволюционные алгоритмы обоснованно можно отнести к разряду систем, способных к самообучению.
* Для оценки оптимальности полученного решения в ЭМ используют не изменение абсолютного значения целевой функции относительно предыдущего шага, а сравнивают показатели целевой функции для каждого из элементов популяции относительно друг друга.
* В алгоритмах ЭМ применяют вероятностные методы поиска, а не детерминированные как в задачах математической оптимизации.
* Так как поиск происходит из множества точек и применяются мутации, то возможно выходить из локальных минимумов и находить новые области с улучшенными значениями целевой функции.

В настоящий момент эволюционное моделирование имеет серьезные перспективы для развития и применения при решении практических задач. Свой исследовательский вклад в развитие теоретической бызы и исследований сфер практического применения ЭМ в России связано с рядом передовых научных школ, среди которых выделяют МГТУ им. Н.Э. Баумана, Институт прикладной математики им. М.В. Келдыша, Нижегородский государственный университет, Таганрогский технологический институт и другие.

## Концепция и принципы эволюционного моделирования

В 1966 годы в свет вышла книга L. Fogel, A. Owens, M. Walsh «Artificial Intelligence through simulated evolution», которая в русском переводе была издана в 1969 году под названием «Искусственный интеллект и эволюционное моделирование». Инновационной идеей книги было предложение использовать модель эволюции применительно к области моделирования искусственного интеллекта. В книге изложен новый к тому времени подход, основанный на применении эволюционирующих программ.

В качестве основной модели для задания организма автор использует концепцию конечного автомата, т.е. автомата с конечным числом символов входного алфавита и конечным числом состояний. Новаторская концепция, предложенная автором – заменить моделирование некогорого сложного объекта или системы на моделирование процесса эволюции этой системы.

В качестве преимуществ реализации предложеннной концепции выделим следующие:

* в ходе итераций эволюции происходит оптимизация модели на каждом шаге;
* в результе – на выходе алгоритма ЭМ получают готовую модель близкую к оптимальной;
* возможность параллельного исследования одновременно нескольких вариаций системы;
* за счет использования методов ЭМ для проектирования возможно минимизировать влияние человеческого фактора;
* возможность исследования и проектирования сложных систем с гибким поведением;

Принципы эволюционного моделирования заимствованы из биологических теорий естественной эволюции, разработанных учеными-естествоиспытателями, такими как Ж-Б. Ламарк, Ч. Дарвин, И. Шмальгаузен и др.

В качестве условий, необходимых для организации эволюционного процесса выделяют следующие [1, 2]:

* наследственная изменчивость как предпосылка эволюции;
* борьба за существование как контролирующий и направляющий фактор;
* естественный отбор как преобразующий фактор.

Процесс поиска оптимального решения с применением принципов эволюционного метода можно характеризовать как поиск в некотором пространстве, заполненном возможными решениями задачи, путем исследования и итеративного улучшения решения.

Отметим, что для каждого из этапов эволюции характерны следующие особенности:

* Каждая последующая популяция состоит только из «жизнеспособных» объектов (превышающих порог оптимальности).
* Каждая новая популяция более приспособлена, чем предыдущая.
* Каждое следущее поколение при моделировании эволюции зависит только от предыдущего поколения.
* Процесс естественного отбора моделируется через выполнение процедуры селекции из множества полученных решений на каждом этапе.
* Вероятность перехода в следущее поколение пропорциональна оценке качества полученного решения.
* Решения на последующих этапах (потомки) наследуют характеристики от решений на прошлых этапах (родителей) с некоторой изменчивостью.
* Приспособленность каждого индивида популяции (конкретного решения) оценивается с помощью специальной целевой функции, которую называют фитнес (fitness) функцией. Чем больше значение фитнес-функции, тем большее количество потомков особи появится в последующих поколениях.

## Области применения эволюционного моделирования

Перспективными областями для применения методов ЭМ являются:

* Оптимизация функций многих переменных.
* Решение задач комбинаторной оптимизации (например, задача коммивояжера, задача о рюкзаке и т.п.).
* Моделирование систем с адаптивным поведением.

Для систем с адаптивным поведением характерно накопление иноформации в процессе функционирования. Затем полученные данные используют в целях достижения оптимального состояния системы. Адаптивность делает возможным применение подобных систем в условиях неопределенности, а также в условиях изменчивости внешней среды.

* Эволюционное проектирование – перспективное направление эволюционного моделирования. В рамках эволюционного проектирования проводят исследования применимости эволюционных алгоритмов для проектирования. Мотивацией подобных исследования является возможность перехода от концепций автоматизированного проектирования к автоматическому проектированию систем.
* Направление эволюционных вычислений Evolutionary Art предполагает создание алгоритмов для синтеза произведений искусства. Например, известной задачей для проверки возможностей и оценки качества генетических алгоритмов является воссоздание портрета "Мона Лиза" Леонардо Да Винчи. Различные варианты реализации такой задачи приводятся в большинстве пакетов эволюционного ПО.

# Глава 2. Теория и основы генетических алгоритмов. Стандартные модели ГА. Основные понятия и принципы.

Сосредоточим внимание на понятии и принципах генетических алгоритмов (ГА) как направлении эволюционных вычислений наиболее разработанном и нашедшем применение в решении практических задач. Особенно перспективным является применение ГА для поиска решений задач оптимизации.

## **История исследований генетических алгоритмов.**

Идеи генетического алгоритма, идеи применения концепций популяционной генетики в моделировании и проектировании сложных систем в ученом сообществе связывают с именем американского ученого-исследователя Джона Генри Холланда (John Henry Holland). Сферой исследований и предметом научного интереса Холланда является исследование сложных адаптивных систем с использованием математических моделей и компьютерного имитационного моделирования [4].

Рисунок Джон Холланд

В 1975 году была опубликована его знаменитая научная работа «Адаптация в естественных и искусственных системах» (ориг. название – «Adaptation in Natural and Artificial Systems») [5]. Главной темой монографии является изучение особенностей процесса адаптации в сложных системах. В основном внимание автора сосредоточено на рассмотрении и формализации в форме математической модели процессов адаптации в естественных системах. Составление моделей производится применительно к предметной области экономики, генетики, искусственного интеллекта (искусственных нейронных сетей), что придает исследованию многоаспектный характер.

В качестве теоретической базы генетических алгоритмов Холландом сформулирована и доказанна в математическом стиле теорема о схемах (в англ. упоминается как «Holland's schemata theorem» или «Schemata theory»). Иногда в англоязычной литературе на указанную теорему ссылаются как на основную теорему генетических алгоритмов (в оригинале «Basic GA theory»), что подчеркивает ее особую теоретическую значимость для обоснования генетических алгоритмов.

Значительный вклад в развитие принципов генетических алгоритмов внесли ученики-аспиранты Холланда – Кеннет Де Йонг (De Jong K.A.) и Дэвид Голдберг(Goldberg D.E.).

Исследовательская работа «An analysis of the behavior of a class of genetic adaptive systems» Де Йонга [6] была вынесена на защиту в университете штата Мичиган в 1975 году. Исследование было выполнено под руководством профессора Джона Холланда и защищено в качестве диссертации на соискание ученой степени PhD (доктора философии). На сегодняшний день работа опубликована и находится в общем доступе в сети Интернет на официальном сайте университета Джорджа Мейсона (URL <http://cs.gmu.edu/~eclab/kdj_thesis.html>), профессором которого является Кеннет Де Йонг. В теоретическом аспекте работа во многом использует основы, заложенные Холландом в [6], лишь немного обобщая их. Новизна же данной работы заключена в практической направленности исследования. В работе Де Йонга проводится сравнительный анализ эффективности реализаций моделей генетичекого алгоритма в сформулированных им задачах.

Наибольшую известность в данной области приобрела работа другого ученика Холланда – исследование Голдберга под названием «Genetic Algorithms in Search, Optimization and Machine Learning» [7]. В данной работе автор на основании работ Холланда сформулировал и описал конкретную реализацию генетического алгорима, который сейчас принято называть простым (классическим) генетическим алгоритмом.

## Понятие репродуктивного плана.

В основе функционирования классического генетического алгоритма лежит реализация модели эволюционного процесса, выстроенной Холландом в [6]. В качестве движущей силы эволюционного процесса рассматриваются процессы эволюционного отбора, проходящего по принципу «survival of the fittest». Принцип этот в русском переводе часто приводят как «выживает сильнейший», хотя правильным и более аккуратным следует считать перевод «выживает наиболее приспособленный». Отметим, что в качестве критерия оценки относительной приспособленности в задачах оптимизации обычно используют значение целевой функции.

Рассмотрим один из основных механизмов модели эволюции, сформулированный Холландом в виде алгоритма, который он назвал репродуктивным планом (в оригинале – reproductive plan). Репродуктивный план по Холланду [5, стр. 91] есть алгоритм, который в фиксированное время t действует над небольшим подмножеством B(t) из общего множества структур A. В терминах генетики множество A можно интерпретировать как множество допустимых особей (хромосом), B(t) как некую промежуточную популяцию, а каждый единичный элемент как особь (хромосому). Алгоритм репродуктивного плана может быть описан в виде цикла, на каждом шаге которого производится генерация новых структур путем модификации имеющихся.

В общем виде алгоритм репродуктивного плана по шагам можно описать следующим образом:

1. Выбор единичной структуры из набора с вероятностью пропорциональной оценке эффективности структуры.
2. Копирование выбранной структуры и применение генетических операторов к выбранной структуре для порождения новой структуры.
3. Выбор второй структуры из набора возможных с равной вероятностью для каждой из структур и замена на новую структуру, полученную на шаге 2.
4. Оценка эффективности новой структуры, запись оценки.
5. Возврат к шагу 1.

Таким образом, репродуктивный план по Холланду предполагает процесс модификации структур (особей) в рамках фиксированного набора (популяции). При этом число структур остается неизменным, т.е. число особей в популяции фиксировано. Приведенное описание носит довольно абстрактный характер, что позволяет интерпретировать его с точки зрения различных реализаций модели эволюционных вычислений. Отметим, что описанный алгоритм не накладывает на данном этапе никаких ограничений на содержание генетических операторов. Холланд также приводит более детализованное описание репродуктивного плана. Подробное описание принципа репродуктивного плана, приведенного у Холланда, можно найти в работе Курейчика [8, стр. 52].

## Простой генетический алгоритм

В русскоязычной литературе описанный далее генетический алгоритм называют по-разному: простым, классическим, элементарным; в англоязычной литературе такой алгоритм принято называть «canonical genetic algorithm». Впервые описание простого генетического алгоритма было приведено Голдбергом в исследовательской работе [8]. Голдберг разработал собственную реализацию генетического алгоритма, взяв за основу концепции приведенные в работе своего учителя Холланда [6].

Механизм простого генетического алгоритма (ПГА) практически следует описанным Холландом концепциям, отличаясь лишь деталями реализации. В модели ПГА задействованы основополагающие понятия, описанные Холландом, – репродуктивный план, операторы кроссинговера и мутации.

В общем случае на этапе исследования, предшествующем применению ПГА, необходимо выполнить следующие действия:

* Выбрать способ преобразования (кодирования) исходных данных в последовательность симоволов хромосомы.
* Выбрать вид фитнес-функции, т.е. функции на основании которой будет производиться оценка эффективности особей популяции.
* Определить способ выбора и выбрать элементы начальной популяции.
* Задать необходимые параметры для операторов селекции и модификации популяции.
* Выбрать условие прекращения работы ПГА.

Характерной особенностью применения генетических алгоритмов является особый **способ кодирования** исходных данных. Иноформация, определяющая значимые свойства, должна быть представлена в виде строк фиксированной длины. В простом генетическом алгоритме элементами строки являются двоичные символы. Такие строки принято называть генотипами или хромосомами. По аналогии с элементами ДНК элементы строки принято называть генами. В результате кодирования для применения ПГА значимые свойства должны быть представлены в виде строки двоичных символов.

Применительно к задачам математической оптимизации выбор способа кодирования – задача первостепенной важности. Широкое применение для кодирование генетических признаков, представимых в виде числовых значений, получило представление в виде кода Грея (Gray Code). В частности, в ПГА чаще применяют представление в виде рефлексивного двоичного кода Грея. Отличительной особенностью кодов Грея, делающей его привлекательным для применения в кодировании свойств, является большая гладкость отображения по сравнения с традиционным бинарным отображением. При изменении кодируемого числа на единицу его представление в коде грея изменяется только в одном разряде.

В рамках неклассических генетических алгоритмов используют и другие способы кодирования хромосом – в виде последовательности действительных чисел, последовательности целых числел, последовательности lisp выражений и др.

**Фитнес-функция** (fitness function)– функция, по значениям которой возможно оценить степень приспособленности, "успешности" особи (хромосомы) относительно других элементов заданной популяции. Сама по себе задача подбора фитнес-функции отнюдь нетривиальна, особенно при решении сложных задач. Удачный выбор функции в значительной степени является залогом успешности применения ПГА. Значения полученной функции влияют на отбор особей из генеральной совокупности, которые получают привилегию на размножение. Разница значений фитнес-функции на множестве особей популяции может быть использована в качестве критерия остановки генетического алгоритма.

На предварительном этапе происходит **создание исходной популяции** особей (хромосом). Иногда исходная популяция генерируется случайным образом, также популяция может являться результатом работы другого алгоритма. Применительно к задачам оптимизации сформированная начальная популяция представляет некоторое подмножество допустимых решений задачи.

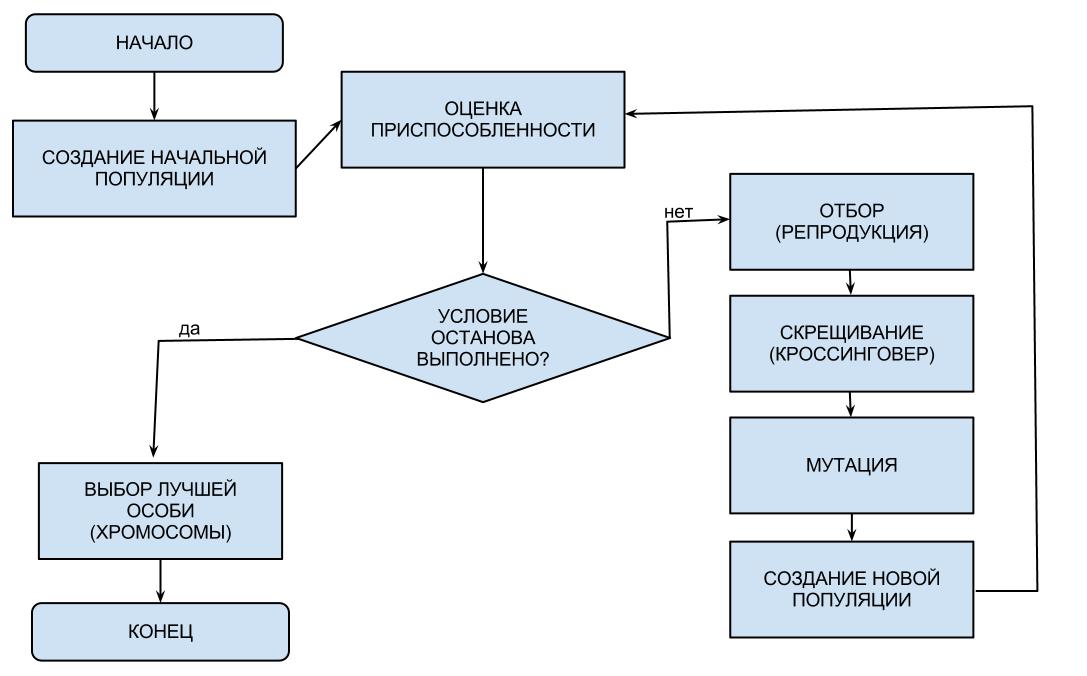


Рисунок Блок-схема простого генетического алгоритма

**Суть алгоритма ПГА** сводится к циклическому процессу, состоящему из следующих шагов:

* Оценка особей текущей популяции с применением фитнес-функции в качестве критерия эффективности отдельной особи популяции.
* Отбор особей из исходной популяции, т.е. формирование некоторого подмножества – промежуточной популяции. Алгоритм отбора определен в виде оператора репродукции (селекции).
* Применение операторов генетической модификации на особях из промежуточной популяции. Для случая ПГА – последовательное применение опрератора кроссинговера (скрещивания) и оператора мутации.
* Обновление состава текущей популяции, например, путем замены особей прежней популяции на модифицированные особи.

На вход каждого цикла ПГА подается набор особей-родителей и задается функция оценки эффективности. Затем из заданного набора особей (текущей популяции) производят выбор особей для промежуточной популяции. В промежуточную популяцию попадают особи (хромосомы), которые получают возможность размножения. В ПГА вероятность репродукции особи пропорциональна значению функции приспособленности. Этим достигается большая вероятность передачи лучших признаков потомству. Из особей промежуточной популяции в результате последовательного применения серии операторов модификации получают новые особи-потомки. На следующем после отбора этапе к популяции применяют последовательно генетические операторы модификации –оператор кроссинговера (скрещивания) и оператор мутации. Особи, полученные в результате, с некоторой заданной вероятностью заменяют особи из начальной популяции.

В ПГА в качестве операторов, применяемых для отбора и модификации особей исходной популяции, используют следующие:

* Оператор репродукции (по Холланду входит в состав понятия «reproductive plan»);
* Оператор кроссинговера (по Холланду оператор «Crossing-over»);
* Оператор мутации (по Холланду оператор «Mutation»);

Рассмотрим немного подробнее каждый из перечисленных операторов генетической модификации, применяемых в ПГА.

**Оператор репродукции** предназначен для выбора из текущей популяции особей с целью последующего проведения размножения их и скрещивания. В литературе оператор также называют опетатором селекции (см., например, [10]). В своих работах Холланд описывает процесс репродукции в рамках введенного им понятия репродуктивного плана. Именно в понятие репродуктивного плана Холланд и вкладывает в качестве функциональной основы репродукцию.

В случае ПГА в качестве метода репродукции используют пропорциональную селекцию. Под пропорциональной селекцией понимают отбор, при котором «число случаев, когда особь выступает в качестве родителя, в среднем пропорционально отношению ее приспособленности к сумме приспособленностей всех особей популяции» [10].

01.gif

Алгоритм пропорциональной селекции, также часто называют селекцией методом рулетки (roulette wheel selection).



Рисунок 3 Селекция методом рулетки

Выбор особи можно представить, как выбор сектора рулетки. Причем, радианная мера секторов установлена в соотношении приспособленности особи к сумме приспособленностей всех особей.

Рулетка в данном случае выступает в роли физической модели. Можно предположить, что процесс моделирования выбора особи в данном случае выглядит следующим образом:

* Каждой особи присваивают уникальный идентификтор (например, особи нумеруют от 1 до N, где N - число особей популяции).
* Колесо рулетки размечают на сектора так, что число секторов равно числу особей в популяции. Размер секторов выбирают пропорционально относительной приспособленности особи в популяции. В результате каждому сектору рулетки возможно уникально сопоставить с идентификатором особи в популяции.
* Раскручивают колесо рулетки. После остановки колеса фиксируют номер выбранного сектора, в соответсветсвии с номером выбирают особь из популяции.

В модификациях генетического алгоритма используют и другие методы отбора, наиболее известными из которых являются линейное ранжирование, метод турнирного отбора, равномерное ранжирование.

**Оператор кроссинговера**

Основной функцией оператора кроссинговера (crossing-over по Холланду) является синтез новых особей популяции на основе отобранных.

Оператор кроссинговера, согласно классической работе Холланда [5, стр. 98] устроен следующим образом:

* Из промежуточной популяции случайным образом выбирают две структуры и
* Далее выбирают целое число x случайным образом из множества индексов {1, 2, 3, ..., }
* Две новые модифицированные структуры формируются из выбранных и посредством обмена подмножествами свойств, расположенными правее относительно свойства с индексом x.

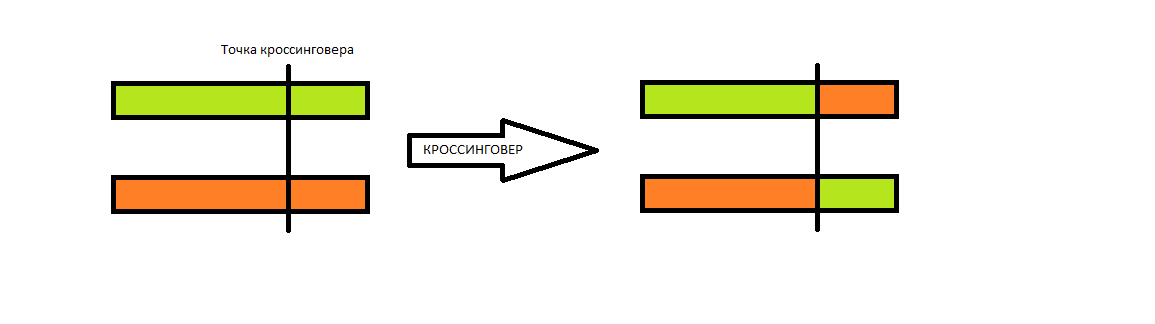


Рисунок Схема одноточечного кроссинговера

Приведенный алгоритм, как видно из описания, предполагает обмен информацией (значениями свойств) между двумя выбранными с некоторой вероятностью хромосомами. Для обмена используется одна точка, определяющая место разделения хромосомы, потому такой оператор называют одноточечным.

Применительно к задачам поиска оптимального решения оператор кроссинговера выполняет функции, направленные на поиск улучшения в областях близких к уже найденным решениям.

Наряду с одноточечным в генетических алгоритмах применяют и другие виды оператора кроссинговера:

* многоточечный кроссинговер;
* метод, основанный на числах Фибоначчи;
* кроссинговер, основанный на дихотомии;
* другие методы.

**Оператор мутации.** После применения оператора кроссинговера с некоторой заданной вероятностью над хромосомами нового поколения выполняют операции мутации. В общем случае процесс мутации заключается в случайном изменении значений генов хромосомы.

Согласно Холланду [6] оператор мутации рассматривается как второстепенный, и значение параметра, задающего его вероятность, целесообразно выбирать достаточно малым.

Существует множество разновидностей реализации указанного оператора. Чаще всего используют мутации в виде перестановки генов хромосомы, различные виды инверсии генов, мутации замены значения гена на случайное значение.

Стоит отметить, что поиск лучших решений не является основной задачей мутации. Функция мутации – выход из локального экстремума и предотвращение преждевременной сходимости.

**Условия останова** цикла выбирают, исходя из поставленной задачи. Например, в качестве условий завершения цикла ПГА могут быть выбраны:

* ограничения на max числа поколений, т.е. число итераций ПГА;
* ограничения по времени работы ПГА;
* достижение необходимого уровня оптимальности;
* схождение популяции (convergence).

# Глава 3. Обзор программных продуктов для генетических вычислений.

Необходимым составляющим этапа реализации в жизненном цикле разработки программного продукта является выбор технологии и инструментария разработки. Объектом исследования данной главы являются доступные библиотеки функций и программные каркасы (фреймворки) для реализации принципов эволюционных вычислений, в том числе генетических алгоритмов.

Отметим, что использование фреймворка позволяет значительно сократить трудоемкость проектирования, по сути предоставляя каркас для реализации методов. При использовании пакетов в виде программных библиотек отпадает необходимость в реализации методов, и наоборот стоит задача компоновки продукта из реализованных методов.

## **Выбор критериев сравнения**

Выбор критериев является значительным для проведения обоснованного сравнения программнных продуктов. С одной стороны он должен достаточно полно отражать основные аспекты и особенности программного комплекса, чтобы сделать возможным оптимальный выбор из множества альтернатив. С другой стороны он не дожен давать слишком подробно сведения, незначительные с точки зрения выбора, чтобы не усложнять процесс принятия решения. Необходимо сформулировать набор критериев и исключить по возможности избыточные характеристики, оставив наиболее значимые.

Первое, что всегда интересует исследователя – функционал, доступный при использовании инструмента. За меру функциональных возможностей примем разнообразие доступных видов представления генов и хромосом, разнообразие генетических операторов – репродукции, скрещивания, мутации, наличие возможностей визуализации и т.п.

Вторым немаловажным моментом является язык программирования. Для целей студенческого исследования ставится условие минимизации времени на освоение технологии программирования, чтобы больше времени уделить изучению самих генетических вычислений. Также для быстрого освоения необходимо наличие и доступность качественной документации и обучающих примеров. Полезным было бы и наличие специализированных форумов, где можно было бы получить ответы на возникшие вопросы.

Еще одним моментом, который необходимо учесть, является тип лицензии программного продукта. В целях исследования необходимо, чтобы лицензия позволяла использовать, изменять и распространять исходные коды продукта. Данное исследование не является коммерческим, поэтому необходимо подобрать программный комплекс, распостраняемый под бесплатной или недорогой лицензией.

Подводя итог, сформируем краткий список основных критериев сравнения:

* возможности кодирования генов;
* набор генетических операторов;
* язык разработки;
* доступность и качество документации;
* уровень поддержки сообщества;
* тип лицензии.

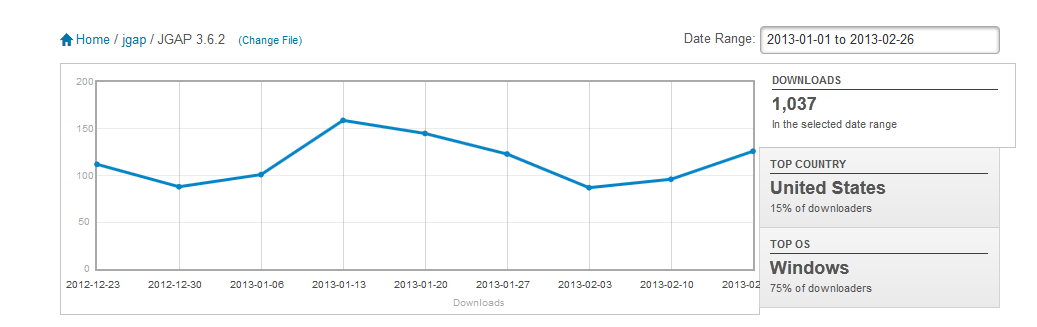
## JGAP



Рисунок Логотип JGAP

JGAP – один из наиболее известных пакетов методов и классов для разработки методов эволюционных вычислений, написанных на языке программирования Java. Исходные коды программ и сопутствующая документация общедоступны и распространяются в сети Интернет [10].

JGAP (Java Genetic Algorithms Package) – бесплатное, свободное программное обеспечение, распространяемое под лицензией GNU Lesser General Public License (GNU LGPL) для некоммерческого использования или под лицензией Mozilla Public License (MPL) для использования в коммерческих проектах.

Поддержка проекта осуществляется интернет-сообществом разработчиков. На момент написания текущий стабильный релиз JGAP имел номер 3.6.2. . Проект JGAP пользуется популярностью у разработчиков, согласно статистике системы управления версиями пакет за год (к моменту исследования) был загружен 1037 раз, при этом 154 загрузки только за последнюю неделю.

В пакете JGAP реализованы большинство необходимых операторов для организации генетического алгоритма:

* различные виды кодирования генов – в виде: целых чисел – IntegerGene, действительных чисел – DoubleGene, булева типа – BooleanGene, символов алфавита – StringGene, бинарных символов – FixedBinaryGene, смешанного типа – CompositeGene;
* операторы селекции реализуют набор различных стратегий отбора: селекция по методу рулетки – WeightedRouletteSelector, ранговая селекция – BestChromosomeSelector, турнирная селекция – TournamentSelector и др.;
* несколько видов операторов мутации: оператор случайной мутации генов – MutationOperator, оператор Гауссовой мутации – GaussianMutationOperator для вещественного представления генов;
* операторы скрещивания: для одноточечного кроссинговера – CrossoverOperator, AverageCrossoverOperator; специальный тип скрещивания с применением "жадного" алгоритма – GreedyCrossover.

В состав пакетов JGAP включен минимальный набор необходимых компонентов для организации генетического алгоритма. В комплекте поставляется несколько примеров реализации и набор обучающих материалов.

## **Watchmaker framework**

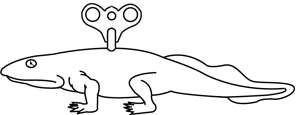


Рисунок Логотип Watchmaker framework

Watchmaker Framework – сравнительно молодой, но активно развивающийся проект, предоставляющий программный каркас (фреймворк), т.е. набор шаблонов для реализации генетических алгоритмов.

Исходные коды и документация проекта доступны в сети Интернет ([]), распространяются под лицензией Apache Software Licence. Исходные коды Watchmaker Framework написаны на языке программирования Java. Поддержка проекта осуществляется небольшой группой разработчиков-энтузиастов. Лицензия продукта дает пользователю право использовать программное обеспечение для любых целей, свободно распространять и изменять содержимое, за исключением названия.

В рамках фреймворка для реализации генетических алгоритмов поставляется следующий набор шаблонов:

* шаблоны для различных стратегий селекции, например, операторы ранговой селекции – RankSelection, TruncationSelection; селекции по методу рулетки – RouletteWheelSelection; селекция с заданным вероятностным распределением – StohasticUniversalScaling; опрератор турнирной селекции – TournamentSelection; возможна организация стратегии элитизма;
* шаблоны для операторов скрещивания на основе механизма кроссинговера с фиксированным числом точек. Доступны следующие шаблоны операторов скрещивания для различных типов представления данных: BitStringCrossingover, ByteArrayCrossover, CharArrayCrossover, DoubleArrayCrossover, IntArrayCrossover, ListCrossover, ListOrderCrossover, ObjectArrayCrossover, StringCrossover;
* шаблоны реализации механизмов мутации для различных вариантов предстваления генов: StringMutation, ListOrderMutation, BitStringMutation.

Стоит отметить, что документации к данному проекту в открытом доступе на данный момент не очень много. Для быстрого освоения программного продукта Watchmaker недостаточно обучающих материалов и пособий в свободном доступе.

## ECJ

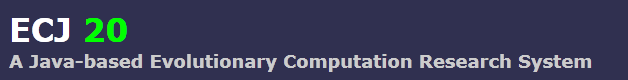


Рисунок Логотип ECJ

ECJ – полнофункциональная система для исследования генетических алгоритмов. Подробную информацию об ECJ можно найти на домашнем сайте проекта [12]. Проект был разработан на базе лаборатории эволюционных компьютерных вычислений (ECLab) при университете Джорджа Мейсона (George Mason University's). Стоит отметить, что Кеннет Де Йонг (ученик Холланда) является одним из основателей ECLab, профессором университета Джорджа Мейсона.

Весь набор программного обеспечения, включенного в состав ECJ, полностью написан на языке программирования Java. Исходные коды ECJ открыты и доступны для загрузки любым желающим. ECJ распространяется под лицензией Academic Free Licence v.3, позволяющей вносить изменения в исходный код и распространять измененные копии. По условиям лицензии необходимо обеспечить доступность исходного кода приложения.

В рамках ECJ реализован широкий набор классов для реализации непосредственно генетических вычислений – генетических алгоритмов, методов генетического программирования.

Основные функциональные особенности ECJ:

* возможности предстваления хромосом с использованием векторов различных типов – ec.vector.IntegerVectorSpecies, ec.vector.FloatVectorSpecies, ec.vector.GeneVectorSpecies;
* реализованы основные виды операторов кроссинговера – одноточечный (one-point), двухточечный (two-point), равномерный (uniform) кроссинговер. Особенно интересны неклассические виды скрещивания, такие как линейное (line) и промежуточное (intermediate) скрещивание, основанные на выборе из подпространства между двумя хромосомами;
* доступны различные виды операторов случайной мутации для каждого из типов представления генов в хромосоме (boolean, integer, Gene); равномерная (uniform) мутация, гауссова (Gaussian) мутация, полиномиальная (polinomial) мутация;
* реализованы специальные виды представления и методы для генетического программирования (GPNodes, GPTrees, GPProblem и др.);
* возможна реализация параллельных генетических алгоритмов на базе островной модели (island model) генетического алгоритма.

В наборе программ ECJ есть все необходимое для реализации полноценного исследования генетических алгоритмов. В сети Интернет доступен достаточный объем документации и других обучающих материалов – электронные версии книг, статьи с примерами реализаций, справочная информация о методах и т.п.

Разработаны и доступны для загрузки дополнительные наборы методов, предназначенных для разработки GUI ECJ графических интерфейсов, построения графиков сходимости эволюционных алгоритмов.

## **Выводы**

Рассмотрев набор программ согласно выбранным критериям возможно сделать выбор наиболее подходящей программы для целей данного исследования. Из рассотренных вариантов явное конкурентоное преимущество для серьезных научных исследований остается за пакетом ECJ. Основные достоинства ECJ – полнота средств реализации генетических алгоритмов, достаточное количество документации и обучающих материалов, возможности интеграции с графическими средствами. Однако, пакет ECJ предназначен для серьезного исследования и вряд ли подойдет новичку в данной области.

Идеальным вариантом для освоения основ генетических алгоритмов является пакет JGAP. В отличие от паета ECJ, пакет JGAP является более простым в освоении и направлен на практическую реализацию методов генетических алгоритмов. Несмотря на его простоту, в пакете JGAP есть все необхоимые для этого элементы и достаточное количество доступной документации.

# Глава 4. Реализация и применение генетических алгоритмов

В предыдущих главах исследования были изложены некоторые аспекты теории генетических алгоритмов и сделан обзор существующих программ для их реализации. Целью исследования данной главы является выработка практических навыков применения генетических алгоритмов. В рамках исследования, применяя методы пакета JGAP, реализуем вариант генетического алгоритма и применим его к решению классической задачи коммивояжера.

## Описание задачи коммивояжера

Задача коммивояжера – классическая задача в области комбинаторной оптимизации. Многим уже известна данная задача, однако для исключения разночтений мы все же приведем ее формулировку. Задача о коммивояжере формулируется следующим образом: «коммивояжер, выходящий из какого-нибудь города желает посетить (n-1) городов и вернуться к исходному пункту. Известны расстояния между всеми этими городами. Требуется установить, в каком порядке он должен посещать города, чтобы общее пройденное расстояние было минимальным» [13]. Вычислительная сложность задачи в случае прямого перебора решений возрастает со скоростью факториала. Помимо чисто интеллектуального применения, решение задачи коммивояжера нашло множество практических применений – в задачах прокладки электропроводов, задачах оптимизации компьютерных программ, задачах прокладки маршрутов и т.д. В поиске эффективных способов решения задачи принимали участие выдающиеся математики прошлого – Л. Эйлер, У. Гамильтон. Работы более современных математиков Д. Данцига и Р. Беллмана также посвящены этой проблеме. В результате проведенных исследований были разработаны математические методы решения задачи, ставшие сейчас традиционными.

Среди общепризнанных методов эффективного решения задачи коммивояжера выделим:

* методы динамического программирования;
* методы линейного программирования;
* эвристические подходы;
* метод Монте-Карло;
* метод ветвей и границ.

Особенно отметим, что применение полного перебора требует чрезвычайных вычислительных затрат при большом числе городов. Применение же простых в реализации эвристических подходов, таких как жадный алгоритм, зачастую приводит к неудовлетворительным результатам.

Довольно перспективным с точки зрения решения задачи коммивояжера представляется применение генетических алгоритмов. Применяя генетические алгоритмы возможно сократить вычислительную сложность задачи, сократив область поиска. В результате возможно получить решение близкое к оптимальному за ограниченное время.

Исходными данными для решения задачи коммивояжера, достаточными для постановки задачи, являются:

* набор городов, обязательных к посещению;
* расстояния между заданными городами.

В общем случае расстояния могут быть заменены на другую произвольную функцию, зависящую от городов. К примеру, в качестве такой функции может быть выбрана стоимость проезда между городами , время пересылки между пунктами и т.п. Решением поставленной задачи в классическом случае является нахождение оптимального маршрута обхода городов.

# Программная реализация генетических алгоритмов

Постановка задачи.

Требуется разработать программу для решения задачи коммивояжера. Целью разработки программного продукта является демонстрация возможностей применения генетических алгоритмов для решения задач комбинаторной оптимизации. В качестве алгоритма для нахождения оптимального маршрута обхода необходимо использовать генетический алгоритм.

Пользователь должен иметь возможность настраивать как параметры исходных данных, так и параметры генетического алгоритма. В результате работы программа выводит точки, символизирующие города, и вычерчивает оптимальный маршрут их обхода.

Особенности реализованного генетического алгоритма.

При формулировании задачи коммивояжера в форме генетического алгоритма необходимо выбрать форму кодирования исходной информации. В данной реализации был выбран способ кодирования в форме строки целых чисел (integer). Каждое из чисел хромосомы представляет номер города из списка к посещению. Длина хромосомы определяется числом городов, которые коммивояжер должен посетить. Порядок посещения городов коммивояжером задается порядком следования номеров им соответсвующих в хромосоме.

На этапе селекции выбор особей из популяции происходит по средством отбора заданного числа особей с наилучшими показателями функции приспособленности. Для проведения скрещивания используется специальный жадный алгоритм скрещивания из пакета JGAP, реализованный специально для задачи коммивояжера. При использовании оператора жадного скрещивания все номера в хромосоме остаются различными, изменяется лишь порядок следования. Для проведения мутации также используется специальный оператор, совершающий случайные перестановки генов в хромосоме.

Краткое описание интерфейса программы.

На рисунке 8 приведено изображение экрана реализованной программы до запуска процесса эволюции. Изображение на рисунке разделено на области, обозначенные цветными рамками с нумерацией.

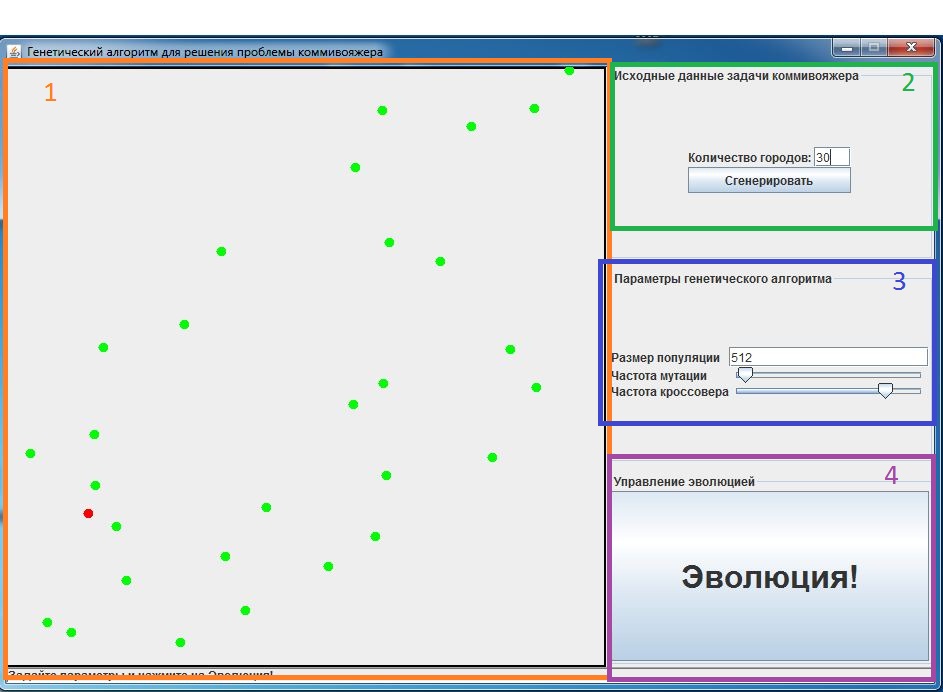
Поясним назначение каждой из указанных на рисунке областей:

Рисунок Интерфейс программы

**1. Область отрисовки результата.** В данной области будет выведен результат работы программы, т.е. наиболее оптимальный маршрут. Зелеными точками обозначены города посещения коммивояжера, красная точка – город из которого коммивояжер начинает маршрут.

**2.Область настройки параметров задачи коммивояжера.** В данной области пользователь имеет возможность настроить число городов обхода. При этом координаты городов выбираются случайным образом. После нажатия кнопки "Сгенерировать" параметры модели задачи обновляются и на экране отображается новая схема расположения городов.

**3. Область настройки параметров генетического алгоритма.** Пользователь имеет возможность выбора основных параметров генетического алгоритма – частоты мутации, частоты кроссинговера, числа особей в популяции.

**4. Кнопка запуска эволюции.** После задания всех параметров нажатием кнопки "Эволюция!" запускается процесс эволюции.

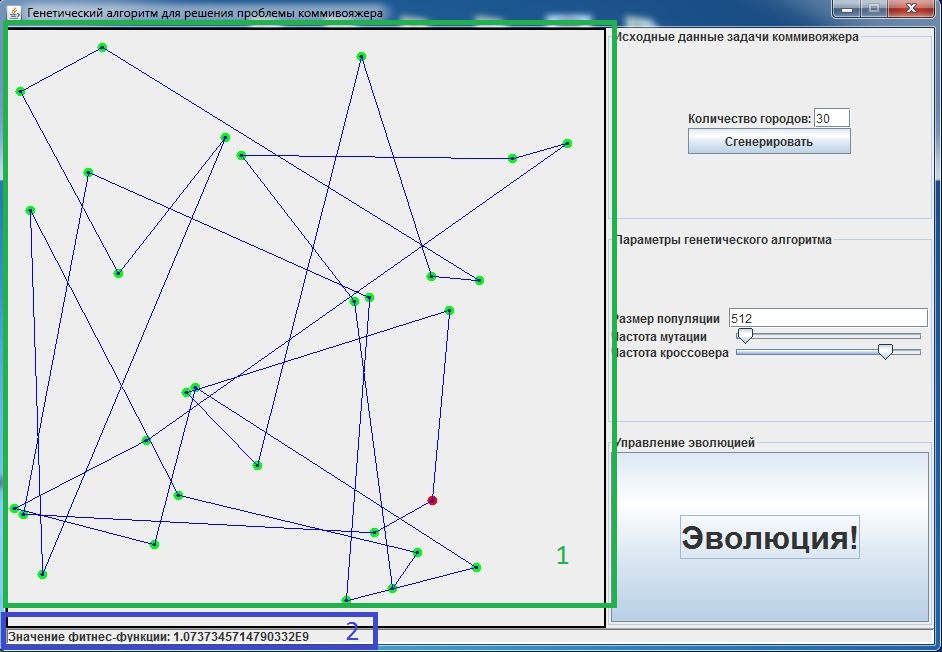


Рисунок Окно программы после запуска

По завершение работы алгоритма в поле 1(см. рис.9) будет отрисован маршрут, который является оптимальным. В строке статуса (поле 2 на рис.9) будет выведено значение фитнес-функции для полученного решения.

# Заключение

Эволюционное моделирование – перспективное направление развития технологий искусственного интеллекта. В перспективе применение принципов эволюционного моделирования может привести к созднанию новых систем с поведением, соответствующим изменяющимся по времени условиям.

Генетические алгоритмы, как одна из областей эволюционных вычислений, уже успела себя отлично зарекомендовать при решении задач многокритериальной оптимизации. Особенно важным преимуществом генетических алгоритмов является возможность сокращения вычислительных затрат при решении задач с общирной областью поиска.

Принципы эволюционного моделирования сформулированы и известны довольно давно. Имеющиеся к тому времени аппаратные ресурсы ограничивали их применимость в практической деятельности. Предпочтение отдавалось детерминированным методам. На сегодняшний день перпрективным направлением является разработка систем с гибким адаптивным поведением. Применяемые ранее алгоритмы не пригодны для организации адаптивных систем. В сочетании с возможностями современной компьютерной техники эволюционный вычисления имеют широкие перспективы развития.

# Список используемой литературы

1. Казаков П.В. Эволюционное моделирование и его применение. Ознакомительная лекция. – БГТУ, 2009
2. Казаков П.В. Основы искусственного интеллекта: учеб. пособие/ П.В. Казаков, В.А. Шкаберин. – Брянск: БГТУ, 2007. – 196 с.
3. Эволюционное моделирование и генетические алгоритмы: [Электронный ресурс] URL: <http://www.uchi-it.ru/3/1/12.html>
4. Информация о профессоре Д. Холланде: [Электронный ресурс]. // Департамент психологии Мичиганского Университета. URL: <http://www.lsa.umich.edu/psych/people/directory/profiles/faculty/?uniquename=jholland>
5. John H. Holland Adaptation in Natural and Artificial Systems. MIT Press, 1992, ISBN 0262581116
6. DeJong K.A. An analysis of the behavior of a class of genetic adaptive systems // Ph.D. dissertation, Univ. Michigan, Ann Arbor, MI, 1975
7. Goldberg, D.E Genetic Algorithms in Search, Optimization and Machine Learning. Addison-Wesley, 1989
8. Курейчик В.М. Генетические алгоритмы и их применение. – Таганрог: Изд-во ТРГУ, 2002 – 242 с.
9. Еремеев А.В. Генетические алгоритмы и оптимизация. (учебное пособие)
10. Домашний сайт проекта JGAP: [Электронный ресурс] URL: <http://jgap.sourceforge.net>
11. Домашний сайт проекта Watchmaker Framework: [Электронный ресурс] URL: [http:/watchmaker.uncommons.org](http://watchmaker.uncommons.org)
12. Проект ECJ: [Электронный ресурс] // Лаборатория ECLab при университете Джорджа Мейсона. URL: <http://cs.gmu.edu/~eclab/projects/ecj/>
13. Мудров В.И. Задача о коммивояжере. – М.: Издательство «Знание», 1969 – 61 с.

|  |  |  |  |
| --- | --- | --- | --- |
| **Дата** | **ФИО** | **Подпись** | |
| «\_\_\_\_»\_\_\_\_\_\_\_\_\_\_201\_\_г. | Записочный В.В. |  | |
| **Дата** | **ФИО** | **Оценка** | **Подпись** |
| «\_\_\_\_»\_\_\_\_\_\_\_\_\_\_201\_\_г. | Тятюшкина О.Ю.. |  |  |