МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное автономное образовательное учреждение высшего образования   
**«Национальный исследовательский   
Нижегородский государственный университет им. Н.И. Лобачевского»**

**(ННГУ)**

**Институт информационных технологий, математики и механики**

**Кафедра: Математического Обеспечения и Суперкомпьютерных Технологий**

Направление подготовки: «Фундаментальная информатика и информационные технологии»

Магистерская программа: «Компьютерная графика»

**МАГИСТЕРСКАЯ ДИССЕРТАЦИЯ**

**Тема:**

**«Генетические алгоритмы оптимизации в потоковых задачах»**

Допущена к защите Выполнил:

Заведующий кафедрой: студент группы 381806-2м

Доронин Роман Олегович

ученая. степень, ученое звание, ф.и.о. ф.и.о.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ подпись подпись

Научный руководитель:

к.ф.-м.н.

Шапошников Дмитрий Евгеньевич

ученая степень, ученое звание, ф.и.о.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

подпись

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

ученая степень, ученое звание, ф.и.о.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

сь

Нижний Новгород  
2019

# ОТЧЕТ Обзор области

**Приведем ряд определений основных понятий, которые используются для описания генетических алгоритмов.**

Поколение ­– это совокупность особей, обособленная от других групп, с представителями которых потенциально возможен генетический обмен.

Индивид (особь) – единичный представитель популяции.

Хромосома – это упорядоченные последовательности генов, то есть структуры, содержащие генетическую информацию об индивиде.

Ген – это атомарный элемент генотипа, в частности, хромосомы.

Генотип – это набор хромосом данной особи. Следовательно, особями популяции могут быть генотипы либо единичные хромосомы (в довольно распространенном случае, когда генотип состоит из одной хромосомы).

Фенотип – это набор значений, соответствующих данному генотипу, то есть декодированная структура или множество параметров задачи.

Аллель – это значение конкретного гена.

Локус – позиция размещения данного гена в хромосоме (цепочке).

## Генетический алгоритм

Ниже описана основная разновидность генетического алгоритма, подробное описание принципов их работы можно найти, в работе [6, 7].

Генетический алгоритм — это эвристический алгоритм поиска. Наиболее часто он используется для решения задач оптимизации и моделирования путём случайного подбора, комбинирования и вариации искомых параметров с использованием механизмов. Называется алгоритм генетическим, так как напоминает биологическую эволюцию и является разновидностью эволюционных вычислений. Подробно генетические алгоритмы освещены в работах J.R.Koza [7].

Задача генетического алгоритма – максимизация или минимизация функции приспособленности. Эта задача решается постепенно – по мере эволюции. Эволюция состоит из последовательности поколений – популяций.

Отличительной особенностью генетического алгоритма от эволюционной стратегии является акцент на использование оператора «скрещивания», который производит операцию рекомбинации решений-кандидатов.

Для каждого поколения отбираются особи с наилучшими значениями приспособленности. Хромосомы отобранных особей с некоторой вероятностью подвергаются операции скрещивания, затем, с другой вероятностью, подвергаются мутациям, а вслед за этим переносятся в следующее поколение. Обычно вероятность мутации намного меньше вероятности скрещивания. Отбор, скрещивание и мутация являются генетическими операциями, их реализация зависит от конкретной задачи. Размер популяции и функция приспособленности не изменяются в процессе эволюции от поколения к поколению.

Кратко сформулировать принцип генетического алгоритма можно следующим образом: наименее приспособленные особи вымирают, а наиболее приспособленные выживают и дают потомство. Предполагается, что каждое последующее поколение будет содержать “более приспособленных” особей, нежели предыдущее.

Таким образом, основные этапы эталонного генетического алгоритма:

1. задание функций приспособленности для особей популяции для решаемой задачи (для каждой задачи подбирается своя функция приспособленности);
2. создание начальной популяции;
3. этап генерации решения;

* применение генетических операций;
* вычисление значение целевой функции для всех особей в поколении;
* формирование нового поколения (селекция);
* выбор “лучшей” хромосомы;
* проверка условий останова;

На предложенном ниже рисунке схематически представлена общая схема работы генетического алгоритма (рис. 2):

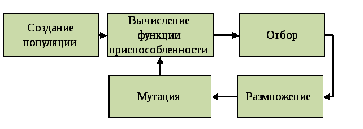


Рис. 2. Общая схема работы генетического алгоритма

По сравнению с традиционными методами оптимизации генетические алгоритмы имеют следующие преимущества:

* генетические алгоритмы легко модифицируются для параллельных вычислений;
* легко приспосабливаются для решения широкого спектра задач;
* хорошо подходят для оптимизации недифференцируемых функций;

Основными недостатками генетических алгоритмов являются:

* высокая трудоемкость;
* сложность контроля генерации решения;
* сложность оценки степени пригодности конкретных генетических алгоритмов для решения конкретной задачи.

Обоснование функционирования генетических алгоритмов приводится в работе [7] и более формализованное (правда с существенными ограничениями на функцию приспособленности) приводится в работе [8]. Математическая модель традиционного генетического алгоритма описана в работах [9, 10].

# Генетическое программирование

Генетическое программирование – разновидность генетических алгоритмов, в которой вместо низкоуровневых представлений объектов в виде битовых строк используются деревья разбора, диаграммы переходов конечного или клеточного автомата и т.д. С помощью генетического программирования наиболее эффективно решаются задачи автоматического построения некоторых классов задач, конечных автоматов и клеточных автоматов.

Стагнация генетического алгоритма – такое состояние алгоритма, при котором на протяжении большого числа поколений не было изменения лучшего значения функции приспособленности у популяции, но текущее решение сильно отличается от глобального минимума.

Схождением называется такое состояние популяции, когда все хромосомы популяции имеют много схожих частей и находятся в области некоторого экстремума. В такой ситуации операции скрещивания практически никак не могут изменить популяцию. А вышедшие из этой области за счет мутации особи склонны вымирать, так как чаще имеют значение функции приспособленности, которое хуже, чем в остальной популяции. Если данный экстремум является глобальным минимумом, то это означает, что найдено лучшее или близкое к нему решение.

Вырождением популяции называется такая разновидность схождения, когда текущий экстремум сильно отличается от глобального минимума, то есть почти вся популяция забилась решениями, которые удовлетворяют какому-либо локальному минимуму. Вырождение популяции является одной из основных проблем при использовании генетических алгоритмов.

## Стандартные операции скрещивания

Общая часть для всех алгоритмов следующая: особи промежуточной популяции случайным образом разбиваются на пары. Каждая из них с вероятностью pc скрещивается, т. е. к ней применяется оператор кроссовера, в результате чего получаются два потомка. Они записываются в новое поколение. Если же паре не выпало скрещиваться, в новое поколение записываются сами особи этой пары.

Точечное скрещивание: выбираются пары хромосом из родительской популяции. Далее для каждой пары отобранных таким образом родителей разыгрывается позиция гена (локус) в хромосоме, определяющая так называемую точку скрещивания – lk. Если хромосома каждого из родителей состоит из генов L, то очевидно, что точка скрещивания lk представляет собой натуральное число, меньшее L. Поэтому фиксация точки скрещивания сводится к случайному выбору числа из интервала [1, L-1]. В результате скрещивания пары родительских хромосом получается следующая пара потомков:

* потомок, хромосома которого на позициях от 1 до lk состоит из генов первого родителя, а на позициях от lk + 1 до L – из генов второго родителя;
* потомок, хромосома которого на позициях от 1 до lk состоит из генов второго родителя, а на позициях от lk + 1 до L – из генов первого родителя.

Двухточечное скрещивание: отличается от точечного скрещивания тем, что родительские хромосомы обмениваются участком генетического кода, который находится между двумя случайно выбранными точками скрещивания.

Многоточечное скрещивание представляет собой обобщение предыдущих операций и характеризуется соответственно большим количеством точек скрещивания.

# Стратегии отбора нового поколения

Выделяют два типа формирования нового поколения после получения множества детей в результате генетических операций:

* новые особи замещают родителей;
* новое поколение составляется из совокупности и новых особей, и их родителей, например, выбором определенного числа лучших.

Ниже рассмотрим основные используемые стратегии отбора нового поколения.

1. Стратегия рулетки. При использовании такого метода вероятность выбора хромосомы определяется ее приспособленностью, то есть чем лучше значение функции приспособленности у хромосомы, тем выше у нее шанс попасть в новое поколение
2. Турнирный отбор. Его суть заключается в том, что случайно выбирается несколько особей из популяции и победителем выбирается особь с наибольшей приспособленностью, она и попадает в новое поколение.
3. Принцип элитизма. В новое поколение обязательно включается заданное количество лучших особей предыдущего поколения (часто только одна лучшая особь).

У каждого и описанных выше методов есть свои преимущества и недостатки, поэтому чаше всего используется их комбинация. Так использование стратегии элитизма часто оказывается весьма полезным для эффективности, так как не допускается потеря лучших решений, то есть полученное хорошее решение будет оставаться в популяции до тех пор, пока не будет найдено решение, которое лучше находящегося в популяции. Недостаток использования стратегии элитизма в том, что повышается вероятность попадания алгоритма в локальный минимум.

## Модели генетических алгоритмов

Очевидно, что использование только специализированной стратегии отбора не всегда помогает избежать локальных экстремумов в процессе генерации решения. Поэтому зачастую вместо стандартного генетического алгоритма используют разнообразные модели, которые в некоторых случаях улучшают сходимость процесса. Ниже рассмотрены наиболее используемые модели генетических алгоритмов, описаны их основные преимущества и недостатки.

* Параллельные ГА

Очевидно, что в генетическом алгоритме присутствуют множество независимых вычислений, следовательно, можно организовать несколько параллельно выполняющихся процессов для увеличения производительности на многопроцессорных устройствах. Ниже приведем несколько возможных стратегий распараллеливания генетического алгоритма:

1. Параллельное выполнение генетических операций и алгоритмов отбора нового поколения. Например: параллельный турнирный отбор: заведем N/2 процессов, каждый из них будет выбирать случайно из популяции 4 особи, проводить 2 турнира, и победителей скрещивать. Полученные дети будут записываться в новое поколение. Таким образом, за один цикл работы одного процесса будет сменяться целое поколение.
2. Параллельное вычисление значений фитнесс функций у особей. Заводятся N процессов. Особи, у которых необходимо выполнить вычисления кладутся в специальную очередь. Каждый процесс вытаскивает особь из очереди и производит необходимые вычисления.

* Островная модель

Эта модель является одной из разновидностей параллельных генетических алгоритмов, однако кроме распределения вычислительной нагрузки она часто применяется для решения задач, для которых характерна многоэкстремальность функции приспособленности.

Принцип работы стандартной островной модели заключается в том, что вместо одной большой популяции используется набор популяций меньшего размера. Эти популяции развиваются независимо, и только изредка происходит обмен представителями между популяциями.

Так как чаще всего число особей, содержащихся в островных популяциях невелико, то такие подпопуляции склонны к преждевременной сходимости. Поэтому важно правильно установить частоту миграции. Чересчур частая миграция или миграция слишком большого числа особей приведет к смешению всех подпопуляций, тогда островная модель будет несильно отличаться от обычного ГА. Если же миграция будет слишком редкой, то она не сможет предотвратить преждевременного схождения подпопуляций. Также заметим, что иногда при миграции производится мутация или какая-либо иная модификация перемещаемых особей.

Генетические алгоритмы стохастичны, поэтому при разных его запусках популяция может сходиться к разным решениям. Островная модель позволяет запустить алгоритм сразу несколько раз и пытаться совмещать «достижения» разных островов для получения в одной из подпопуляций наилучшего решения. Использование островной генетической модели позволяет во многих случаях избежать локальных минимумов, что положительно сказывается на сходимости к решению. Заметим, что при этом зачастую время работы генератора несколько больше, чем у стандартной модели, в силу того, происходит дупликация генерации решений.

* Клеточная модель генетических алгоритмов

Эта модель представляет собой вариацию островной модели. На этапе инициализации строится двумерная сетка. В каждой ячейке сетки запускается содержится особь, при этом определяется процесс взаимодействия между ячейками: каждая ячейка может взаимодействовать только с четырьмя своими соседями (сверху, снизу, слева, справа), при этом для нее выбирается лучшую особь среди своих соседей, скрещивается с ней особь из своей ячейки и полученный ребенок помещается в свою ячейку вместо родителя. Заметим, что в ячейке может находиться не одна ячейка, а подпопуляция, тогда описанная выше модель представляет собой вариацию островной модели, в которой накладываются ограничения на взаимосвязи между островами.

По мере работы такой модели возникают эффекты, похожие на островную модель. Сначала все особи имеют случайную приспособленность. Спустя несколько поколений образуются небольшие области похожих особей с близкой приспособленностью. По мере работы алгоритма происходит рост и конкуренция областей между собой.

Основное преимущество заключается в том, что происходит генерация сразу нескольких решений в разных «областях». За счет этого использование клеточной генетической модели позволяет во многих случаях избежать локальных минимумов. Основной недостаток заключается в более низкой скорости сходимости по сравнению со стандартным генетическим алгоритмом.

# Постановка задачи

Во многом генетические алгоритмы похожи на классические методы оптимизации, популяция — это набор текущих точек, мутации — это исследование соседних точек, отбор — это выбор новых точек для поиска решения в условиях ограниченных вычислительных ресурсов.  
  
Популяция всегда стремится к ближайшему максимуму, так как мы отбираем текущие точки поиска, как имеющие максимальное значение (все остальные точки «умрут», не выдержат конкуренции с ближайшим максимумом). Так как размер популяции значительный, а значит вероятность сделать хотя бы один шаг в направлении максимума не пренебрежимо мала, то через некоторое количество шагов популяция сместится в сторону локального максимума. А потомки точки, смещенной ближе к максимуму, имеют большую «выживаемость». Значит через достаточное количество шагов, потомки этой точки начнут доминировать в популяции и вся она сместится к максимуму.

# Обзор технологий

В последнее время все больше «ходят» разговоры про новомодные алгоритмы, такие как нейронные сети и генетический алгоритм. Сегодня я расскажу про генетические алгоритмы, но давайте на этот раз постараемся обойтись без заумных определений и сложных терминах.   
Как сказал один из великих ученных: «Если вы не можете объяснить свою теорию своей жене, ваша теория ничего не стоит!» Так давайте попытаемся во всем разобраться по порядку.

### Щепотка истории

Как говорит Википедия: «Отец-основатель генетических алгоритмов Джон Холланд, который придумал использовать генетику в своих целях аж в 1975 году». Для справки в этом же году появился Альтаир 8800, и нет, это не террорист, а первый персональный компьютер. К тому времени Джону было уже целых 46 лет.

### Где это используют

Поскольку алгоритм самообучающийся, то спектр применения крайне широк:

* Задачи на графы
* Задачи компоновки
* Составление расписаний
* Создание «Искусственного интеллекта»

### Принцип действия

Генетический алгоритм — это в первую очередь эволюционный алгоритм, другими словами, основная фишка алгоритма — скрещивание (комбинирование). Как несложно догадаться идея алгоритма наглым образом взята у природы, благо она не подаст на это в суд. Так вот, путем перебора и самое главное отбора получается правильная «комбинация».   
Алгоритм делится на три этапа:

* Скрещивание
* Селекция (отбор)
* Формирования нового поколения

Если результат нас не устраивает, эти шаги повторяются до тех пор, пока результат нас не начнет удовлетворять или произойдет одно из ниже перечисленных условий:

* Количество поколений (циклов) достигнет заранее выбранного максимума
* Исчерпано время на мутацию

##### Более подробно о технологиях

##### С++ библиотека компонентов генетических алгоритмов На этом шаге создается начальная популяция, которая, вполне возможно, окажется не кошерной, однако велика вероятность, что алгоритм эту проблему исправит. Главное, чтобы они соответствовали «формату» и были «приспособлены к размножению».

##### Библиотека для реализации генетических алгоритмов в .NET Framework 2.0. Ну тут все как у людей, для получения потомка требуется два родителя. Главное, чтобы потомок (ребенок) мог унаследовать у родителей их черты. При это размножаются все, а не только выжившие (эта фраза особенно абсурдна, но так как у нас все в сферическом вакууме, то можно все), в противном случае выделится один альфа самец, гены которого перекроют всех остальных, а нам это принципиально не приемлемо. Мутации. Мутации схожи с размножением, из мутантов выбирают некое количество особей и изменяют их в соответствии с заранее определенными операциями. Отбор. Тут начинается самое сладкое, мы начинаем выбирать из популяции долю тех, кто «пойдет дальше». При этом долю «выживших» после нашего отбора мы определяем заранее руками, указывая в виде параметра. Как ни печально, остальные особи должны погибнуть.