МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ

ГОСУДАРСТВЕННОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ  
ВЫСШЕГО ПРОФЕССИОНАЛЬНОГО ОБРАЗОВАНИЯ

НОВОСИБИРСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ

Кафедра Автоматизированных Систем Управления   
 (полное название кафедры)

Ермошенко Павел Андреевич

(И., О., фамилия студента – автора работы)

Разработка программного обеспечения для оптимизации маршрута обхода

(полное название темы магистерской диссертации)

геометрических объектов на плоскости на основе метода колонии муравьев

**МАГИСТЕРСКАЯ ДИССЕРТАЦИЯ**

обзор источников

##### Руководитель

Фроловский

Владимир Дмитриевич  
(фамилия, И., О.)

д.т.н., профессор

(уч. степень, уч. звание)

Новосибирск, 2015 г.

# 1. Введение

В данной работе будет рассмативаться задача обхода контуров геометрических объектов на плоскости. Рассматриваемая задача является более частным случаем широко известной задачи коммивояжера (Traveling Salesman Problem, TSP). Задача TSP заключается в том, чтобы найти оптимальный маршрут через набор “городов” (представленных точками на графе), проходящий через каждый город хотя бы раз и вернуться в исходный город. Вариации задачи могут включать полный (коммивояжер может отправиться в любую точку напрямую из любой другой) или неполный графы. Сама задача относится к классу NP-трудных. В большинстве случаев для ее решения применяются итеративные оптимизационные алгоритмы (прямой перебор становится невозможным уже при 66 городах), одним из которых является “алгоритм муравьиных колоний”.

Решение задачи обхода геометрических фигур на плоскости может помочь в оптимизации некоторых областей промышленности. Так, например, лазерный резак, движущийся по оптимальному маршруту во время резки использует существенно меньше как запаса прочности своих деталей, так и энергии, необходимой для его перемещения. На производстве, занимающемся нарезкой больших объемов деталей экономия, полученная при оптимизации может играть большую роль.

Для построения маршрута обхода геометрических объектов на плоскости, одной из возможных стратегий является выделение всех вершин фигур в набор точек, а также задание некоторых “обязательных маршрутов” (граней графа, которые резак должен пройти обязательно) для прохождения. Также, нужно выбрать некую стратегию по добавлению дополнительных точек на гранях фигур с целью создания между ними путей перехода от одной фигуре к другой. Создание самого программного продукта потребует выбора некоторых технологий программирования.

Рассмотрим некоторые источники информации по данным вопросам.

# 2. Источники

## 2.1. Ant **Colony** Optimization (Marco Dorigo, Thomas Stutzle).

Сложное социальное поведение муравьев давно изучается наукой. В данный момент, известно, что эти шаблоны поведения могут служить моделью для решения сложных оптимизационных задач в информационных технологиях. Попытка разработать алгоритмы, вдохновленные одним из аспектов поведения муравьем, способностью находить кратчайшие маршруты, вылилась в область исследований под названем “Алгоритм Муравьиной Колонии” (Ant Colony Optimization, ACO), одного из известнейших и самых успешных методов оптимизации. В этой книге описана быстро растущая область с ее теоретического начала до практического применения. Она также включает множество доступных ACO алгоритмов и их применения. В начале, книга описывает перевод наблюдаемого поведения муравьев в работающие оптимизационные алгоритмы. Далее вводится и обсуждается метаэвристика муравьиной колонии. Она рассматривается в контексте задач оптимизации. Затем следует детальное описание и руководство ко всем основным ACO алгоритмам, а также отчет по нынешним теоретическим находкам. Книга описывает практические применения ACO алгоритмов, включающие в себя построение маршрутов, машинное обучение, планирование и задачи биоинформатики. В деталях описан алгоритм под названием AntNet, созданный для решения проблемы маршрутизации в вычислительных сетях. Книга завершена подведением итогов по продвижению в научной области и описанием будущих направлений исследований.Каждая глава заканчивается библиографическим материалом, включающим список самых важных идеи, представленных в главе, а также упражнениями. Алгоритм Муравьиной Колонии будет являться предметом интереса академических и промышленных исследователей, стремящихся реализовать этот алгоритм на практике.

Один из авторов этой книги, Марко Дориго является главным исследователем FNRS (Belgian National Funds for Scientific Research) и одним из создателей IRIDIA, лаборатории искусственного интеллекта университета “Université Libre de Bruxelles”. Он является изобретателем метаэвристики муравьиной колонии. Область его нынешних исследований включает групповой интеллект, групповую робототехнику и метаэвристику для дискретной оптимизации. Он является главным редактором журнала “Swarm Intelligence” (Групповой Интеллект), а также редактором множества журналов о вычислительной технике и адаптивных системах. Доктор Дориго был награждет “Italian Prize for Artificial Intelligence” в 1996 году, “Marie Curie Excellence Award” в 2003, “Dr. A. De Leeuw-Damry-Bourlart” в 2005, “Cajastur "Mamdani" International Prize” в 2007, а также “ERC Advanced Grant” в 2010.

## 2.2. Ant Colony Optimization (Utkarsh Jaiswal, Shweta Aggarwal).

Научная работа под названием “Ant Colony Optimization” раскрывает еще несколько областей применения уже известного алгоритма. Эта статья включает в себя краткое описание истории научных конференций посвященных ACO. Оптимизационный алгоритм рассматривается как часть большей области – Искусственного Интеллекта. ACO относится к области ИИ, которая называется “Групповой Интеллект”. В то время как ИИ применяется во многих областях, можно выделить три основных класса подходящих задач:

* “Обычные”: зрение, речь, понимание человеческой речи, способность осознавать окружающий мир
* “Формальные”: игры (шахматы, шашки), математика (геометрия или логика), доказательство свойств программ (правильность)
* “Экспертные”: инженерия (дизайн, изготовление), планирование, научный анализ

В статье данная классификация представлена более подробно в виде дерева. Далее статья описывает методы сравнения систем группового интеллекта, а также их основные принципы. Перечислены некоторые принципы этих систем:

* Самоорганизация (базируется на обратной связи, случайных составляющих и множественном взаимодействии)
* Стигмергия (косвенные коммуникации посредством воздействия на окружающую среду)

В статье также представлены достоинства и недостатки группового интеллекта. Статья завершается перечислением нынешних (Задача коммивояжёра и задачи планирования) и описанием возможных будущих применений алгоритма ACO.

Впервые статья была опубликована в журнале “International Journal of Scientific & Engineering Research” в июле 2011 года. Авторами этой статьи являются Utkarsh Jaiswal (университет неизвестен) и Shweta Aggarwal (студентка New Delhi Institute of Management).

## 2.3. Scheduling of flexible manufacturing systems: an ant colony optimization approach (R Kumar, M K Tiwari and R Shankar).

Данная статья описывает задачу планирования для FMS (Flexible Manufacturing Systems – Гибкие Производственные Системы). Такие системы состоят из некоторого количества контролируемых машин с автоматической системой обработки материалов, соединенных в компьютерную сеть. Системы называются гибкими так как позволяют решить задачу многими способами (маршрутами). В таком сценарии важно выбрать машину, которая решит задачу быстрее всего с минимальными затратами. Полное решение такой задачи в теории позволит достичь значительной экономии в ресурсах (в том числе временных). Постановка задачи сводится к определению задачи (процесса, выполнение которого нужно запланировать), ее параметризации (неких параметров, позволяющих варьировать процесс), этапов процесса (каждый из которых может выполнятся на отдельной машине) а также доступных машин и их особенностей (аналог параметров процесса).

В статье приводится попытка решения данной задачи путем применения оптимизационного алгоритма муравьиных колоний. Задача представляется в виде графа (это обязательно для применения ACO), в котором узлы являются машинами, которые выполняют части процессов (представляемых путешествующими муравьям – реализацией ACO). Статья предоставляет результаты в виде графиков зависимости качества финального решения от количества итераций ACO.

В заключении статьи приводится вывод об эффективности применения ACO для данной проблемы. Как и множество задач, для которых применялся алгоритм ACO, данная задача получила “достаточно хорошее” решение с его помощью. В большинстве случаев в реальных условиях, получение лучшего возможного решения не представляется возможным (в силу ограничения вычислительных ресурсов), в то время как решение, полученное итеративными методами обычно не сильно отличается от лучшего, но может быть получено с использованием намного меньшего количества ресурсов.

## 2.4. Ant Colony Optimization routing algorithm for selecting Multiple Feasible Paths for Packet Switched Networks (Meenakshi R Patel, Babita Kubde).

Эта статья была опубликована в журнале “IJCSI International Journal of Computer Science Issues” в 2012 году. Эта статья является рецензией на статью, описывающую применение алгоритма ACO в проблеме маршрутизации сети. Применение ACO к проблеме маршрутизации получило отдельное название – AntNet и было предложено оригинальным автором ACO Marco Dorigo в 1996 году.

Основной проблемой, требующей решения в маршрутизации компьютерных сетей является “TCP congestion”. Эта задача состоит в том, как максимально эффективно использовать сетевые ресурсы (мощность и пропускную способность маршрутизаторов), чтобы удовлетворить нужды клиентов, подключенных к сети. Помимо ACO для этой задачи применяется множество алгоритмов “машинного обучения”, результаты были встроены в во множество доступных комплектов сетевого взаимодействия операционных систем. В исходной статье описывается применение ACO, для нахождения параметров TCP congestion. Задача определяется характеристиками маршрутизаторов в сети (узлами графа ACO). Решением будут являться наборы параметров для каждого маршрутизатора, которые позволят достичь лучшей пропускной способности при меньшей нагрузке.

В статье описывается множество вариантов применения ACO к этой задаче (IMACO, antBalance – варианты с множеством колоний). Основной проблемой применения этого алгоритма является так называемая “стагнация”. Стагнация возникает когда большинство муравьев выбирают один и тот же маршрут (основная особенность ACO) и это приводит к уменьшению шанса выбора других маршрутов графа. Такой сценарий в реальных условиях вызовет перегрузку маршрута (при этом другие маршруты будут в состоянии простоя).

Статья завершается выбором алгоритмов множественных колоний как решения проблемы стагнации и возможности оптимизировать сетевую пропускную способность.

## 2.5. An Ant Colony Optimization Algorithm for maximizing the lifetime of Heterogeneous Wireless Sensor Network (Saburi Uday Parvatkar and Deipali V. Gore).

Беспроводная Сенсорная Сеть определяется как сеть небольших встраиваемых устройств (в большинстве случаев, микроконтроллеров), передающих данные друг другу по сети с технологией ad hoc (все устройства в сети равноправны и могут обмениваться данными напрямую с любым другим устройством). Такая сеть предназначена для очень низкой пропускной способности (несколько бит в день) и нацелена на максимизацию времени работы от батареи для каждого индивидуального устройства. Сама задача нацелена на уменьшение стоимости каждого индивидуального устройства (узла сенсорной сети).

Статья описывает уже применяемые в данный момент методы планирования передач для такой сети. Они включают в себя специальные приемы для уменьшения нагрузки на батарею (такие, как ограничение окна передачи во времени и его синхронизация с другими устройствами, таким образом все устройства знают когда нужно ожидать данных и включают режим приема или прослушивания только иногда, вместо того чтобы находиться в нем постоянно). Далее описан подход к минимизации стоимости, включающий использование PSO (Particle Swarm Optimization – оптимизационный алгоритм “Роя Частиц”), использующий некоторый набор частиц, каждая из которых представляет возможное решение задачи. На каждой итерации этого алгоритма частицы имеют некие ускорения, направленные в стороны уже известных лучших решений. Значение целевой функции вычисляется на каждой итерации и ускорения обновляются. Также имеется метаэвристика, предназначенная для предотвращения схода всех точек в одной области (что увеличивает возможность найти новые, более лучшие решения).

Статья завершается описанием применения ACO для этой же задачи и дает описание некоторых его преимуществ по отношению к планированию времени передачи.

## 2.6. Ant colony optimization with re-initialization (Matej Ciba, Ivan Sekaj).

Множество проблем могут быть сведены к графу (что позволит применить ACO). В реальных ситуациях не всегда резонно находить лучшее возможное решение. Вместо этого, в большинстве случаев “достаточно хорошее” решение предпочтительно. Данная статья фокусируется на описании возможных модификаций алгоритма муравьиных колоний и их эффектов.

В обычной ситуации баланс между использованием предпочтительных путей и исследованием новых (так называемое “многообразие путей”) в ACO достигается введением глобального правила обновления феромона (имитирующего испарение со временем). Помимо такого механизма контроля был предложен ввод полностью случайного выбора следующего узла вместо обычного способа (случайная составляющая совмещенная с количеством феромона на ребре графа). При таком выборе многообразие путей возможно даже при ситуации, где один маршрут полностью доминирует остальные по количеству феромона.

Следующим подходом к увеличению многообразия путей является введение лимита на феромон на каждом доступном пути. Таким образом, разность между минимальным и максимальным значением феромона никогда не превышает некое выбранное значение. Так как это значение имеет прямое отношение к выбору муравьем маршрута, такое ограничение уменьшает максимально возможную долю количества феромона в этом выборе.

Статья также описывает подход ре-инициализации работы алгоритма для увеличения многообразия путей. Процедура ре-инициализации состоит в изменении количества феромона в некоторой изменяемой по размеру области. Изменение производится в сторону уменьшения с использованием формулы, включающей среднее арифметическое значение феромона по графу. Это изменение делается таким образом, чтобы не потерять уже полученную информацию о графе.

Статья приводит результаты, которые показывают, что данные модификации улучшают результаты в тех ситуациях, где задача, как и ее лучшее решение изменяется со временем.

## 2.7. Applying Ant Colony Optimization Algorithms to Solve the Traveling Salesman Problem

Этот источник является страницей на популярном ресурсе Code Project. Code Project является веб-сайтом, на котором легко найти достаточно простые для понимания примеры реализации проектов на различных языках программирования. Ресурс позволяет скачать исходный код проектов, а также каждая страница проекта содержит подробное описание принципов проекта и сопровождается частями исходного кода с коментариями. Также, страница проекта имеет рейтинги и комментарии пользователей ресурса, эти комментарии не редко содержат дополнительную полезную информацию.

Проект, описанный на данной странице является программой на языке С++ с использованием ACOTSP как ядра реализации ACO. Также в список используемых библиотек входит MFC (библиотека графического интерфейса от Microsoft), MS Visual Studio (интегрированная среда разработки), Boost (библиотека общего назначения для C++) и GDI+ (библиотека графического интерфейса, позволяющая создавать собственные элементы графики и контроля).

Программа предназначена для расчета кратчайшего пути через все вершины графа и предоставляет способ создания графа и изменения параметров алгоритма через графический интерфейс. Страница проекта содержит математическое описание принципов работы программы, примеры исходного кода, снимки экрана некоторых частей программы в действии и может служить полным руководством к использованию данной программы.

Доступность исходного кода позволяет исследовать реализацию алгоритма оптимизации муравьиной колонии, а описание на странице проекта позволяет скомпилировать и воспользоваться самой программой на компьютере под управлением Windows.

## 2.8. HTML Web API, Canvas Tutorial

В данный момент Web технологии достигли очень широкого распространения в современном мире. Последней версией веб-стандарта является HTML5. Эволюция этого стандарта привела к следующей структуре:

* HTML: язык разметки, используемый для определения структуры документа. В момент выполнения браузер преобразует HTML текст в дерево именуемое DOM (Document Object Model, объектная модель документа).
* DOM: модель документа, которая содержит все элементы документа и их свойства. Представляется в виде дерева.
* JavaScript: язык программирования, позволяющий производить операции над деревом DOM (изменять, удалять и добавлять элементы). Используется для придания странице динамичности (анимаций, логики).
* CSS: язык стилей, используемый для определения внешнего вида элементов DOM. Состоит из так называемых “селекторов” (позволяющих выбрать целевой элемент) и свойств (дающих возможность изменить внешний вид элемента, такой как, например, цвет фона или размер шрифта).

HTML5 является синонимом для совокупности всех вышеперечисленных технологий и является самой распространенной “средой исполнения” в мире. Это обусловлено тем, что практически каждый компьютер имеет веб-браузер.

Одним из элементов HTML5 является Canvas (“полотно”). Он позволяет легко создавать простые графические формы на плоскости. Это можно использовать для создания интерфейсов, позволяющих визуально задавать параметры задачи.

Mozilla Developer Network является основным ресурсом документации по HTML5. Этот ресурс содержится компанией Mozilla, ответственной за один из известнейших браузеров – Firefox. Ресурс содержит подробную документацию по Canvas и другим механизмам, а также позволяет проверить совместимость нынешних браузеров с интересующими пользователя технологиями.

# Список использованных источников

1. Ant Colony Optimization (Marco Dorigo, Thomas Stützle, 2004): книга, печатное издание.
2. Ant Colony Optimization (Utkarsh Jaiswal, Shweta Aggarwal): <http://www.ijser.org/researchpaper/Ant_Colony_Optimization.pdf> [Электронный ресурс] Дата обращения: 01.02.15.
3. Scheduling of flexible manufacturing systems: an ant colony optimization approach (R Kumar, M K Tiwari and R Shankar): <http://www.et.byu.edu/~ered/ME482/Paper_Topics/TomAnt_ColonyOptimization.pdf> [Электронный ресурс] Дата обращения: 01.02.15.
4. Ant Colony Optimization routing algorithm for selecting Multiple Feasible Paths for Packet Switched Networks (Meenakshi R Patel, Babita Kubde): <http://ijcsi.org/papers/IJCSI-9-2-3-271-273.pdf> [Электронный ресурс] Дата обращения: 02.02.15.
5. An Ant Colony Optimization Algorithm for maximizing the lifetime of Heterogeneous Wireless Sensor Network (Saburi Uday Parvatkar and Deipali V. Gore): <http://inpressco.com/wp-content/uploads/2014/06/Paper1792116-2118.pdf> [Электронный ресурс] Дата обращения: 02.02.15.
6. Ant colony optimization with re-initialization (Matej Ciba, Ivan Sekaj): <http://article.sciencepublishinggroup.com/pdf/10.11648.j.acis.20130103.14.pdf> [Электронный ресурс] Дата обращения: 09.02.15.
7. Applying Ant Colony Optimization Algorithms to Solve the Traveling Salesman Problem: <http://www.codeproject.com/Articles/644067/Applying-Ant-Colony-Optimization-Algorithms-to-Sol> [Электронный ресурс] Дата обращения: 09.02.15.
8. HTML Web API, Canvas Tutorial: <https://developer.mozilla.org/en-US/docs/Web/API/Canvas_API/Tutorial?redirectlocale=en-US&redirectslug=Canvas_tutorial> [Электронный ресурс] Дата обращения: 10.02.15.