

Санкт-Петербургский государственный университет

Математико-механический факультет

Кафедра Информационно-аналитических систем

Математическое обеспечение и администрирование информационных систем

Бусаров Вячеслав Геннадьевич

Продуктовая кастомизация посредством моделирования

Магистерская диссертация

Научный руководитель:
канд. ф.-м. н., доцент Графеева Н. Г.

Рецензент:
Технический директор DSLab LLC Шатерников А. Н.

Санкт-Петербург
2019

SAINT-PETERSBURG STATE UNIVERSITY

Mathematics and Mechanics Faculty
Department of Analytical Information Systems
Software and Information Systems Administration

Vyacheslav Busarov

Product Customization by Modeling

Graduation Thesis

Scientific supervisor:
associate professor Natalia Grafeeva

Reviewer:
Chief Technology Officer DSLab LLC Alexey Shaternikov

Saint-Petersburg
2019

Оглавление

Введение	4
Постановка задачи	6
1. Данные	7
1.1. Параметры окружающей среды	7
1.2. Параметры транспортной компании	8
1.3. Параметры транспортного средства	9
2. Обзор	11
3. Оптимизация функции	13
3.1. Выбор алгоритма	13
3.2. CMA-ES	15
4. Кастомизация посредством моделирования	19
5. Эксперименты	20
6. Заключение	23
Список литературы	24

Введение

Продуктовая кастомизация – это процесс, позволяющий предоставить клиенту наилучшее индивидуальное предложение в пределах имеющихся возможностей. На основе одного и того же продукта можно сформировать большой диапазон совершенно различных предложений с разными свойствами, что делает этот процесс значимым для конечного потребителя.

Экспертная оценка становится невозможной даже на небольших примерах: продукт с 10 параметрами, каждый из которых изменяется в 3 возможных значениях, имеет более 59 тысяч конечных версий.

Кроме того, определение свойств может проводиться как с заранее сформированными опциями и их комбинациями, так и на этапе проектирования, когда для отобранных групп пользователей необходимо сформировать наиболее подходящие значения параметров продукта. В данном случае количество вариантов несоизмеримо возрастает, так как характеристики из дискретных величин превращаются в непрерывные.

В продуктах для частного (Business-to-Consumer, b2c) использования имеют значение неизмеримые факторы, такие как статус приобретаемого бренда или эмоции, получаемые при использовании. В коммерческом секторе (Business-to-Business, b2b) эти аспекты исключаются, и решение о покупке принимается, исходя из эффективности продукта. Такой подход диктуется рынком: неэффективные компании не выживают.

Сегодняшние рекомендательные системы используют один и тот же подход к b2c и к b2b сегментам рынка, в то время как в одном секторе ключевым факторам принятия решения являются эмоции, а в другом – экономическая целесообразность. Кроме того, они не принимают во внимание индивидуальные особенности: рекомендация формируется в результате сопоставления клиента с группой других клиентов или продукта с группой других продуктов. Никакого персонального предложения здесь быть не может.

Я предлагаю формировать индивидуальное предложение, непосредственно исходя из деятельности компании-клиента. Рассмотрим процесс эксплуатации продукта как сложную функцию, значением которой является себестоимость.

Для этого выделим настраиваемые параметры продукта, независимые от нас параметры среды эксплуатации и поток задач, для решения которых создаётся продукт. Также выберем ключевой показатель эффективности (чаще всего им будет себестоимость или прибыль). Всё это позволит нам создать модель бизнес-процессов, которую можно рассматривать как сложную функцию (black box function). Её значением станет выбранный ранее показатель эффективности. Тогда наиболее подходящий для клиента продукт будет находиться в точке экстремума построенной функции.

Для подобных функций, не заданных аналитически, существует целый ряд алгоритмов оптимизации. Поскольку в реальности время, отведённое на решение, строго ограничено, критерием выбора станет максимальная точность результата, достижимого за ограниченное количество итераций.

Данный подход к продуктовой кастомизации может быть применён к любому коммерческому продукту, но в данном исследовании я рассмотрю малотоннажный грузовой транспорт (Light Commercial Vehicles) и одну из крупнейших транспортных компаний в России. На реальных данных о её деятельности будут сформированы рекомендации по подбору оптимальных транспортных средств.

Постановка задачи

Цель данной работы — сформировать подход к кастомизации и применить его для решения задачи подбора оптимального малотоннажного транспортного средства.

Для достижения этой цели в рамках данной работы были сформулированы следующие задачи:

- сформировать подход к кастомизации,
- выбрать алгоритм для решения подзадачи оптимизации,
- применить результаты исследования при подборе оптимального транспортного средства.

1. Данные

В данной работе описан процесс кастомизации малотоннажного грузового транспорта (Light Commercial Vehicles) для компании EMS Russian Post. Данные подразделяются на параметры транспортного средства, параметры окружающей среды и параметры транспортной компании и содержат информацию по городу Москва за январь, февраль и июль 2018 года.

1.1. Параметры окружающей среды

Для хранения карта местности используется сервис OpenStreetMap.org. Город разделён на зоны доставки, согласно бизнес-процессу компании. Каждая зона представляет собой замкнутый многоугольник, описанный набором вершин (latitude, longitude) в порядке обхода. Разметка сделана с помощью Google Maps (рис. 1). Всего существует 5 зон, в каждой из которых находится свой склад для доставки и для хранения невостребованных отправок. За подбор заказов в каждой зоне несут ответственность несколько групп курьеров.

Дорожная сеть хранится при помощи сервиса GraphHopper.com и является ориентированным взвешенным связным графом. Все точки, в которых возможно изменения маршрута (поворот, разворот), а также здания представлены вершинами графа, а связывающие их дороги – рёбрами. Вес ребра состоит из нескольких показателей: протяжённость, максимально разрешённая скорость, реальная скорость с учётом конкретной дорожной ситуации. С помощью этого же сервиса происходит построение маршрутов.

Трафик – исторические данные, собранные при помощи сервиса waze.com. Это информация о максимальной скорости на каждом участке дороги (ребре графа дорожной сети) за каждые 30 минут в январе, феврале и июле 2018 года. Кроме того, чтобы не допустить переобучения на конкретной дорожной ситуации, она варьируется на основе заданной вероятности возникновения затора на том или ином участке.

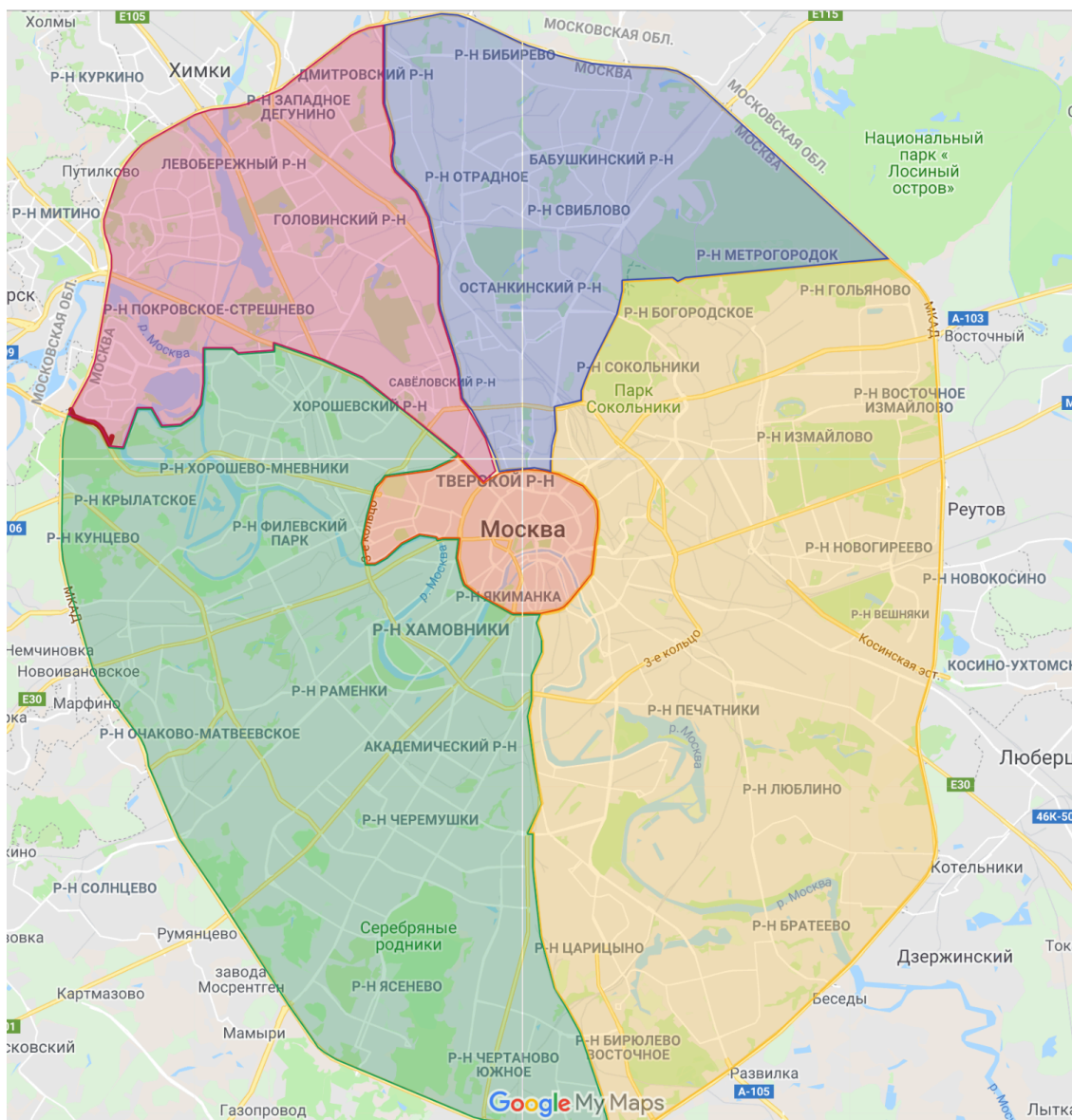


Рис. 1: Зоны курьерской доставки по городу Москва.

1.2. Параметры транспортной компании

Поток заказов задан адресом отправителя, адресом доставки, весом и объёмом отправления, а также временем поступления заказа. Все адреса сопоставлены с конкретными вершинами на дорожной карте. В списке находятся в том числе отменённые и не востребоваанные заказы, таких порядка 12%.

Курьеры описаны посменно. Для каждой смены задано:

- время начала работы,
- время окончания работы,
- стартовая точка,
- обслуживаемая зона,
- количество курьеров.

Для каждой из зон доставки указано месторасположения складов, используемых для заказов последней мили междугородной доставки и для хранения не востребуемых отправок. Ещё один параметр – гарантированное время доставки заказов, которое нельзя превышать.

1.3. Параметры транспортного средства

- тип кузова (body type),
- длина (length),
- высота (height),
- ширина (width),
- длина колёсной базы (wheelbase length),
- ширина колёсной базы (wheelbase width),
- тип передней подвески (type of front suspension),
- тип задней подвески (rear suspension type),
- тип топлива (fuel type),
- объём двигателя (powertrain),
- количество цилиндров (number of cylinders),

- количество турбин (number of turbines),
- наличие задних дверей (rear doors),
- наличие боковых дверей (side door),
- тип привода (drive shaft),
- диаметр колёс (wheel diameter),
- ширина колёс (wheel width),
- мощность двигателя (horsepower),
- ёмкость топливного бака (fuel capacity),
- тип тормозной системы (brake type),
- материал тормозной системы (brake material),
- тип коробки передач (transmission type),
- количество передач (count of speed).

Характеристики транспортного средства (следствие сконфигурированных параметров).

- грузоподъёмность (payload),
- грузовместимость (cargo volume),
- вес (weight),
- расход топлива городском/смешанном цикле (fuel consumption),
- стоимость ТО (maintenance cost),
- стоимость ОСАГО (insurance cost),
- размер транспортного налога (transport tax).

2. Обзор

На сегодняшний день существует два варианта решения задачи кастомизации: экспертная оценка и рекомендательные системы. Экспертная оценка проводится вручную и основывается на знаниях и опыте специалистов. Однако, количество вариаций одного и того же продукта делает такой подход неэффективным. Например, 10 параметров, каждый из которых имеет всего лишь 3 значения, порождает более 59 тысяч комбинаций.

Кастомизацию можно провести при помощи рекомендательных систем, если каждый вариант продукта рассматривать как отдельный продукт, после чего выбирать один из них. Существует две стратегии создания рекомендательных систем: фильтрация, основанная на содержании (content-based filtering), и коллаборативная фильтрация (collaborative filtering)[10][9].

Суть фильтрации по содержанию в выделении критериев выбора продукта и отличительных черт пользователя. Они составляют профили пользователей и продуктов, что позволяет выстраивать ассоциации[11].

Коллаборативная фильтрация использует исторические данные о поведении группы пользователей, из которых выделяются люди с похожими предпочтениями и шаблонами поведения. Данная стратегия базируется на предположении о том, что данному пользователю понравятся продукты, выбранные похожими на него пользователями[14].

В процессе исследования мною была проанализирована 121 статья по тематике рекомендательных систем, из которых 43 использовали стратегию фильтрации, основанной на содержании, 65 – коллаборативную фильтрацию, и 13 – комбинацию данных подходов.

Первая стратегия не подходит для решения задачи кастомизации b2b продуктов, так как единственный критерий в данном случае – себестоимость транспортной работы. Она может быть вычислена только при помощи реального использования продукта или посредством моделирования. Но сделать это для каждого варианта не представляется возможным ввиду вычислительной сложности.

Вторая стратегия использует исторические данные о покупках группы пользователей, но у компаний-клиентов зачастую нет такой информации, либо её слишком мало. Кроме того, разнообразие бизнес-процессов настолько велико, что большинство случаев являются уникальными, что делает использование данного подхода невозможным. Однако, кастомизация посредством моделирования не сталкивается с подобными проблемами.

3. Оптимизация функции

3.1. Выбор алгоритма

На сегодняшний день многие системы действуют как «чёрные ящики». Это упрощает эксплуатацию, но усложняет понимание их работы. Оптимизация подобных систем становится всё более актуальной по мере возрастания их сложности[3]. Когда мы говорим о модели бизнес-процессов реальной компании или об их части, сложность модели становится особенно критичным фактором.

Формально, задана функция $f : X \subseteq R^n \rightarrow R$, для которой можно вычислять значения $f(x), \forall x \in X$ при отсутствии доступа к любой другой информации о функции f .

В большинстве случаев работы с «чёрными ящиками» основные затраты времени приходятся на вычисление очередного значения функции. Поскольку я решаю реальную практическую задачу, время, отведённое на поиск экстремума, ограничено. Все алгоритмы имеют различную точность, что является критически важным для конечного результата. Поэтому, критерий выбора алгоритма – максимальная точность решения при ограниченном количестве итераций. Оно определяется общим ограничением времени на поиск оптимума, а также сложностью вычисления очередного значения функции в рамках имеющихся ресурсов.

Для вычисления точности работы алгоритма оптимизации нам понадобится термин триал (trial) – это набор значений параметров x , который будет использоваться для однократного вычисления значения функции $f(x)$. Триал может быть законченным (completed), что означает наличие вычисленного значения $f(x)$. В противном случае он называется ожидаемым (pending)[5].

Введём несколько контрольных функций $\varphi(x)$ (benchmark functions) с заранее известными неочевидными оптимальными точками. Тогда точность работы алгоритма оптимизации будем оценивать при помощи «разрыва оптимальности» (optimality gap). Если x^* – точка миниму-

ма функции $\varphi(x)$, а x – лучшее решение, найденное оптимизатором, то $|\varphi(x^*) - \varphi(x)|$ – разрыв оптимальности. В случае наличия стохастического компонента внутри алгоритма будет вычислять «средний разрыв оптимальности» (average optimality gap), получаемый в результате многократного запуска для одной и той же контрольной функции[5].

Сравнение между разными контрольными функциями само по себе затруднительно, так как они имеют разные множества значений. Для того, чтобы нормализовать оценку точности, рассмотрим отношение разрыва оптимальности конкретного алгоритма поиска экстремума к значению того же показателя алгоритма случайного поиска (Random Search) для одной и той же контрольной функции.

Контрольные функции для сравнения алгоритмов поиска экстремума по критерию точности результата возьмём из практикума по оценке работы оптимизаторов «чёрных ящиков» (Black-Box Optimization Benchmarking Workshop)[4].

Алгоритмы поиска экстремума функций – чёрных ящиков используют минимальное количество предположений о рассматриваемой задаче, существует множество подобных алгоритмов и их вариаций. Специалисты Google Research провели сравнительный анализ и выявили наиболее производительные оптимизаторы[5]. Это: метод многорукого бандита с использованием регрессии на основе Гауссовского процесса («a multiarmed bandit technique using a Gaussian process regressor») сокращённо GP Bandit[5], SMAC алгоритм[5], стратегия эволюции ковариационной матрицы адаптации («Covariance Matrix Adaptation Evolution Strategy») сокращённо CMA-ES[7], вероятностный метод поиска («probabilistic search method»)[5].

В исследовании [5] проводились эксперименты с разными размерностями. Для данной размерности d были сгенерированы контрольные функции с соответствующим количеством измерений d . Каждый алгоритм оптимизации был запущен для каждой контрольной функции 100 раз, после чего был зафиксирован средний результат по всем запускам.

Рис. 2 отображает результаты экспериментов: по шкале абсцисс отложено количество итераций, по шкале ординат – точность, измеренная

как отношение разрыва оптимальности конкретного алгоритма к значению того же показателя алгоритма случайного поиска[5]. При этом двойным алгоритмом случайного поиска (2xRandom Search) назван алгоритм случайного поиска, на каждой итерации которого вычисляется значение контрольной функции для двух триалов, вместо одного, как у прочих алгоритмов.

Результаты сравнительных экспериментов позволяют выбрать оптимальный алгоритм для разных размерностей и ограничений по времени работы, выраженных в количестве итераций. В моих экспериментах на реальных данных рассматриваемый бизнес-процесс моделируется функцией, переменными которой являются описанные выше параметры транспортного средства. Таким образом количество измерений оптимизируемой функции наиболее близко к 32 при ограничении на количество итераций порядка 10^3 . Исходя из этого, наиболее подходящим является алгоритм CMA-ES.

3.2. CMA-ES

Evolution Strategy (ES)[7] – алгоритм стохастического поиска, который решает задачу минимизации функции – чёрного ящика, действующей из пространства $S \subseteq R^n$ в R . На каждом шаге поиска имеется набор точек – потенциальных точек экстремума оптимизируемой функции, именуемых потомками. Среди них при помощи заранее заданной функции селекции (приспособленности) выбираются лучшие для продолжения работы алгоритма. После чего происходит стохастическая вариация, также известная как «мутация» или «рекомбинация», – это процесс порождения новых потомков на основе уже имеющихся точек. На следующем шаге алгоритма действия повторяются[6].

Мутация осуществляется за счёт добавления вектора нормально распределённых случайных величин. Данное распределение называется мутационным, а его параметры – стратегическими, поскольку именно от них зависит производительность алгоритма[12]. Для стратегических параметров существует вероятность порождения с их помощью точки,

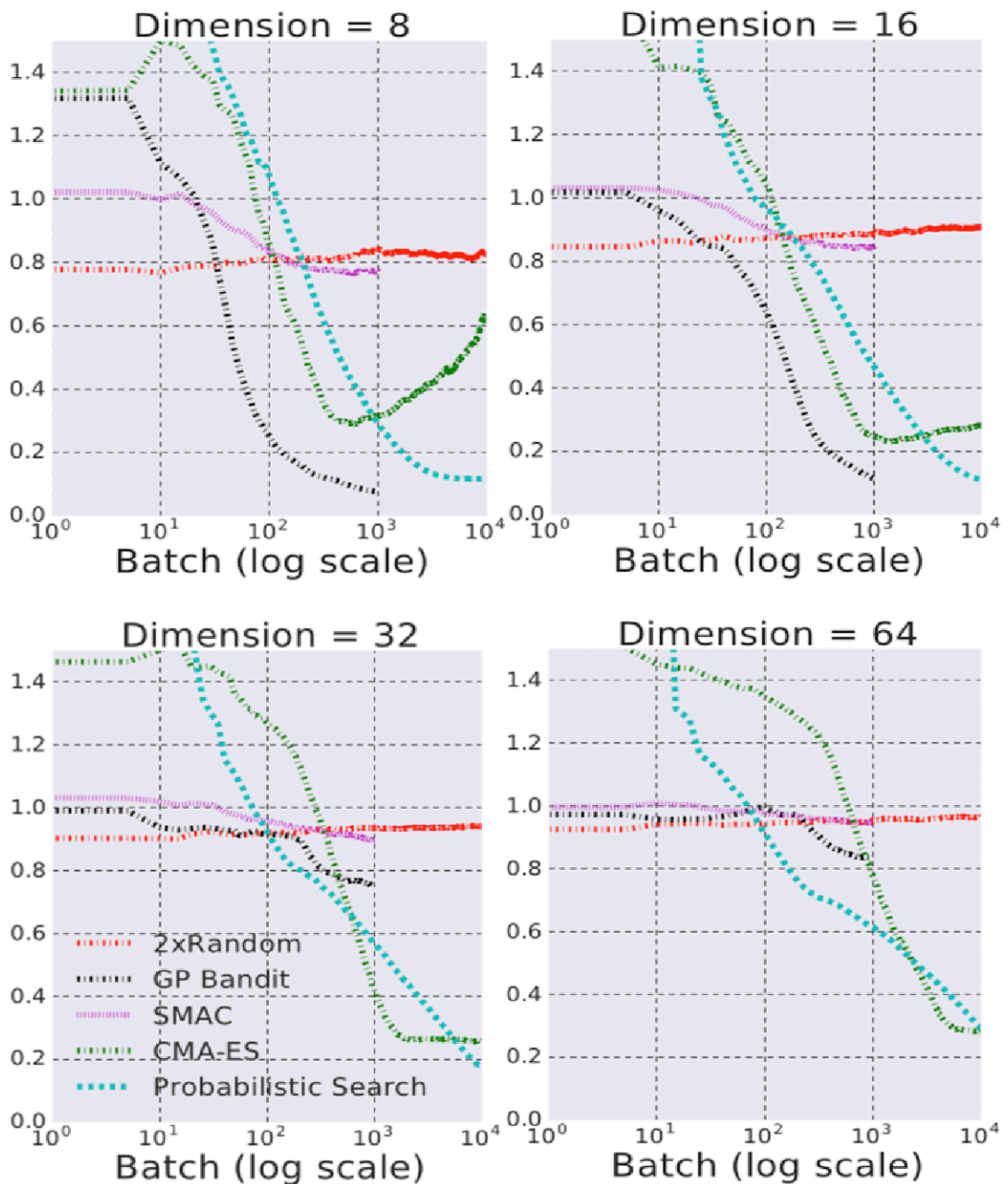


Рис. 2: Эксперименты по сравнению алгоритмов оптимизации функций – чёрных ящиков[5].

которая будет выбрана лучшей на очередном шаге алгоритма. Из этого естественным образом возникает цель – максимизировать эту вероятность.

На каждом шаге алгоритма происходит следующая мутация[7]:

$$\sigma_k^{(g+1)} = \sigma^{(g)} \exp(\xi_k) \quad (1)$$

$$x_k^{(g+1)} = x^{(g)} + \sigma_k^{(g+1)} z_k \quad (2)$$

где

- g – номер шага алгоритма;
- $x^{(g)} \in R^n$ – поколение точек (потенциальные точки экстремума);
- $\sigma^{(g)} \in R^+$ – параметрический вектор (размер шага мутации);
- $k = 1..λ$ – номера потомков;
- $λ$ – размер популяции;
- ξ_k – вектор случайных величин, чаще всего имеющий нормальное распределение, а в алгоритме CMA-ES $P(\xi_k = 0.3) = P(\xi_k = -0.3) = \frac{1}{2}$ [1][13];
- $z_k \sim N(0, I) \in R^n$ – вектор случайных величин с $(0, I)$ – нормальным распределением, где I – единичная матрица.

Мутационное распределение меняется в зависимости от конкретной задачи, а также в процессе работы алгоритма. Мутации, породившие точки, которые были выбраны лучшими на некотором шаге, могут породить лучшие точки и в ближайшем будущем. На основе этого предположения основана идея уменьшения степени случайности (дерандомизация), позволяющая воспроизвести некоторые ранее удачные мутации. Дерандомизация в алгоритме CMA-ES состоит из двух уровней.

Первый уровень вносит следующие изменения в уравнения (1) и (2):

$$\sigma_k^{(g+1)} = \sigma^{(g)} \exp\left(\frac{\xi_k}{d}\right) \quad (3)$$

$$x_k^{(g+1)} = x^{(g)} + \sigma^{(g)} \exp(\xi_k) z_k \quad (4)$$

где $d \geq 1$ – параметр демпфирования.

Это позволяет:

- облегчить индивидуальную адаптацию для каждой задачи, так как позволяет регулировать разницу между $\sigma_k^{(g)}$ и $\sigma_k^{(g+1)}$;
- уменьшить стохастические флуктуации стратегических параметров.

Второй уровень вносит следующие изменения в уравнения (3) и (4):

$$\sigma_k^{(g+1)} = \sigma_k^{(g)} \exp\left(\frac{\|z_k\| - E[\|N(0, I)\|]}{d}\right) \quad (5)$$

$$x_k^{(g+1)} = x_k^{(g)} + \sigma_k^{(g)} z_k \quad (6)$$

Это позволяет:

- явно контролировать скорость изменений;
- увеличить вероятность порождения точек, которые будут выбраны лучшими на очередном шаге.

4. Кастомизация посредством моделирования

В данном исследовании я описываю работу с b2b продуктами, так как в данном сегменте рынка решения принимаются на основе прагматических выводов, а не эмоций, как в b2c[8][2]. Задача – получить продукт, решающий проблемы конкретного бизнеса, с наименьшей совокупной стоимостью владения – TCO (Total Cost of Ownership). В случае транспортных компаний TCO принято вычислять, как стоимость транспортной работы, измеряемую в рублях за перевозку 1 кг на 1 км. При этом мелкие грузы весом до 0.5 кг приравниваются друг к другу. Данный показатель – следствие процесса эксплуатации, значит он может быть вычислен посредством моделирования деятельности компании-клиента. Модель можно рассматривать как «чёрный ящик», то есть функцию, не имеющую аналитического представления. Параметры данной функции описывают деятельность клиента и окружающую среду, а переменные – настраиваемые параметры эксплуатируемого продукта. Таким образом, задача сводится к поиску экстремума сложной функции.

Для всех компаний, работающих в определённой сфере, предполагается наличие одной общей модели, которая после настройки будет имитировать деятельность любой компании. Для сферы внутригородской доставки малотоннажным транспортом это будет модель имитации перевозок из точки в точку (peer-to-peer) и со склада в точку (hub-to-peer). В качестве входных данных модель будет принимать:

- параметры окружающей среды – дорожная сеть, трафик;
- параметры транспортной компании – поток заказов, курьеры, графики работы, склады, гарантированное время доставки;
- параметры используемого транспорта.

5. Эксперименты

Модель построена на платформе AnyLogic и представляет собой мультиагентную имитацию процесса внутригородской доставки по городу Москва, где каждый курьер реализован в виде агента. Поток заказов и параметры транспортной компании – реальные данные рассматриваемой компании. Используется историческая информация о распределении заказов между курьерами, при её отсутствии модель способна сама осуществлять диспетчеризацию. Также реализована функциональность отображения процесса работы курьеров (рис. 3)

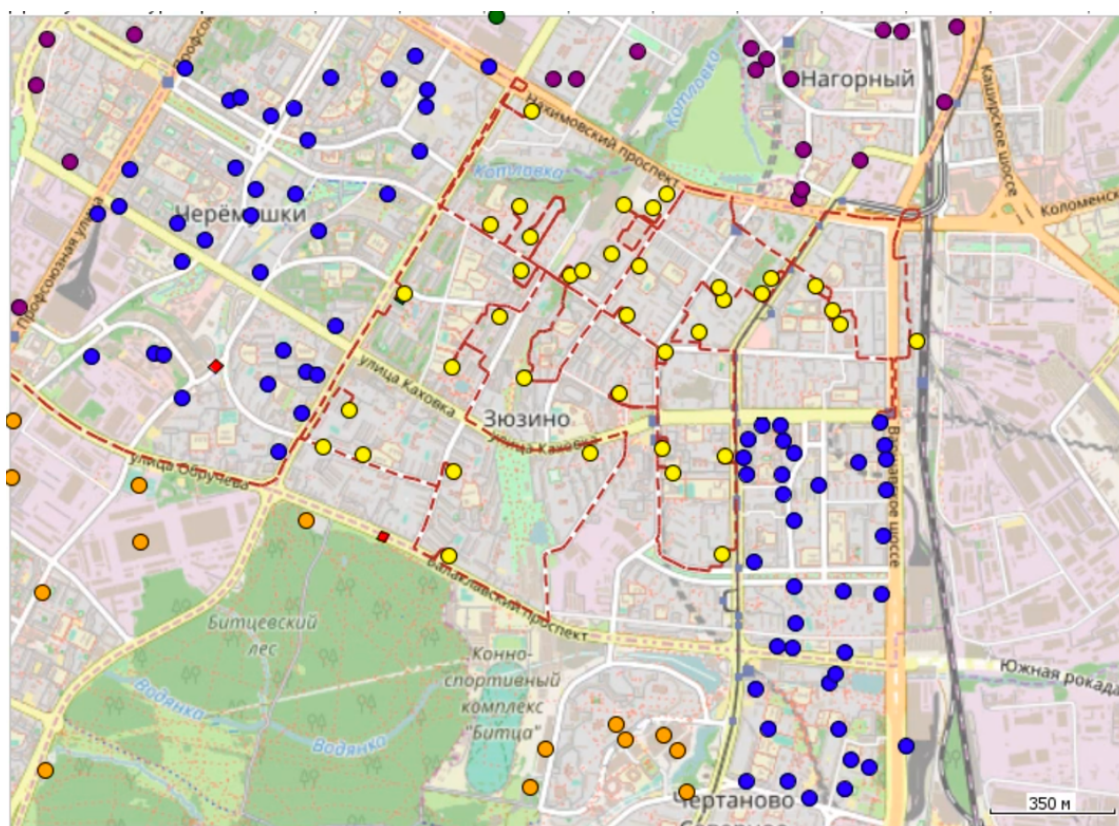


Рис. 3: Пример визуального отображения моделирования работы транспортной компании. Круги разного цвета обозначают заказы, красные ромбы обозначают курьеров, пунктир – маршруты перемещения одного из курьеров.

Значение функции себестоимости доставки вычислялась посредством моделирования, которое проводилось на данных 3-х календарных недель: с 15 по 21 января 2018 года, с 19 по 25 февраля 2018 года, с 16 по 22

июля 2018 года. По имеющимся информации пробег одного транспортного средства в течение года составляет примерно 150 тыс. км. Затраты на техническое обслуживание, страховку и транспортный налог равномерно распределялись на годовой пробег. По результатам моделирования определялась средняя стоимость перевозки 1 кг. на 1 км.

Для оптимизации функции себестоимости использовался алгоритм Covariance Matrix Adaptation Evolution Strategy (CMA-ES)[7]. Переменные оптимизируемой функции – параметры транспортного средства. Его код доступен в open-source проекте Advisor (github.com/tobegit3hub/advisor). Эксперименты проводились на Mac Book Pro с 6-ти ядерным процессором Intel Core i9 4.8GHz, объём оперативной памяти – 32GB. В табл. 1 указаны 15 наиболее экономичных моделей транспортных средств, лучшие конфигурации которых продемонстрировали следующие себестоимости транспортной работы.

Транспортное средство	Себестоимость ($\frac{RUB}{kg \times km}$)
Renault Dokker	9.85
Ford Transit	10.16
Volkswagen Caddy	10.72
Mercedes Sprinter Classic	11.28
Volkswagen Transporter	11.48
Renault Master	12.32
Gazel NEXT	12.66
Volkswagen Crafter	13.84
Renault Kangoo	14.39
Peugeot Expert	14.71
Peugeot Partner	15.13
Lada Largus	15.92
Peugeot Boxer	15.96
Fiat Ducato	15.98
Citroen Jumpy	15.99

Таблица 1: Результаты замеров себестоимости транспортной работы

Себестоимость транспортной работы для рекомендованных нами автомобилей Renault Dokker и Ford Transit составила $9.85 \frac{RUB}{kg \times km}$ and $10.16 \frac{RUB}{kg \times km}$ соответственно. Используемый в настоящее время автомобиль

Лада Ларгус продемонстрировал себестоимость $15.92 \frac{RUB}{kg \times km}$. Таким образом, выбор кастомизированных транспортных средств позволил бы снизить себестоимость внутригородских перевозок на 38%.

6. Заключение

Итак, в настоящей работе была сформирована методика решения актуальной на сегодняшний день задачи кастомизации. Описанный мной подход имеет следующие преимущества перед рекомендательными системами:

- **Индивидуальный результат.** Рекомендательные системы предлагают то, что подошло клиентам со схожей деятельностью. Моё решение предлагает то, что оптимально подходит для данного конкретного клиента и его уникальной деятельности.
- **Быстрый старт.** Может быть большой диапазон различных активностей, отличающиеся по множеству параметров, в том числе не описанных явно. Нынешним рекомендательным системам необходимо накопить базу знаний для начала работы. Моему решению – только получить параметры деятельности компании-клиента.
- **Возможность работать с непрерывными параметрами.** Большинство рекомендательных систем работают с дискретными наборами значений.

Кроме того, в данной работе описан способ выбора наиболее подходящего алгоритма решения подзадачи оптимизации функции – «чёрного ящика».

Важным результатом стало применение подхода на реальных данных крупнейшей в России логистической компании EMS Russian Post, для неё я сконфигурировал и оттранжировал оптимальные версии малотоннажных грузовых автомобилей. Это стало возможным благодаря появлению критерия выбора наиболее экономически выгодных конфигураций автомобилей для транспортных компаний, исходя из данных об их бизнес-процессах.

Список литературы

- [1] Back T., Schwefel H.P. An overview of evolutionary algorithms for parameter optimization. *Evolutionary computation*, vol. 1. — 2016. — P. 1–23.
- [2] Bagozzi R.P., Prashanth M. G. The role of emotions in marketing. *Journal of the academy of marketing science*. — Springer, 1999. — P. 27–184.
- [3] Duarte A., Laguna M., Martí R. Black-box solvers. *Metaheuristics for business analytics*. — Springer, 2018. — P. 105–136.
- [4] Finck S., Hansen N., Rost R. Real-parameter black-box optimization benchmarking 2009: presentation of the noiseless functions. — 2009.
- [5] Golovin D., Solnik B., Moitra S. Google vizier: a service for black-box optimization. *KDD 17*. — 2017. — P. 1487–1495.
- [6] Gong D., Sun J., Miao Z. A set-based genetic algorithm for interval many-objective optimization problems. *Transactions on evolutionary computation*, vol. 22. — IEEE, 2018. — P. 47–60.
- [7] Hansen N., Ostermeier A. Completely derandomized self-adaptation in evolution strategies. *Evolutionary computation 9*, vol. 2. — 2001. — P. 159–195.
- [8] Kemp E.A., Borders A.L., Anaza N.A. The heart in organizational buying: marketers understanding of emotions and decision-making of buyers. *Journal of business and industrial marketing*, vol. 33. — 2018. — P. 18–29.
- [9] Koutrika G. Modern recommender systems: from computing matrices to thinking with neurons. *SIGMOD 18*. — 2018. — P. 1651–1654.
- [10] Koutrika G. Recent advances in recommender systems: matrices, bandits, and blenders. *Openproceedings.org*. — 2018.

- [11] Pazzani M.J., Billsus D. Content-based recommendation systems. The adaptive web. — Springer, 2007. — P. 291–324.
- [12] Rechenberg I. Evolutionsstrategie, optimierung technischer systeme nach prinzipien der biologischen evolution. Frommann-holzboog. — 1993. — S. 157–160.
- [13] Rechenberg I. Optimierung von strukturen. Evolutionsstrategie 94, vol. 1. — 1994. — S. 167–182.
- [14] Schafer J.B., Frankowski D., Herlocker J. Collaborative filtering recommender systems. The adaptive web. — Springer, 2007. — P. 1291–324.