Глобальная сложность задачи составления расписания состоит в гигантском пространстве возможных решений (проклятие размерности) и дискретности. Вместе это означает, что малейшие изменения в решении способные сильно его улучшить или испортить, причём, поскольку мы не действуем в непрерывном пространстве, эти последствия невозможно предсказать, опираясь на градиент или что-то ещё. По этой причине мы будем искать решение с помощью генетических алгоритмов, тем более что их эффективность для схожих задач уже демонстрировалась в работах [Вирсански Э. Генетические алгоритмы на Python / пер. с англ. А. А. Слинкина. – М.: ДМК Пресс, 2020. – 286 с.: ил.] и [МОДЕЛИ, АЛГОРИТМЫ И ПРОГРАММНЫЕ СРЕДСТВА ОБРАБОТКИ ИНФОРМАЦИИ И ПРИНЯТИЯ РЕШЕНИЙ ПРИ СОСТАВЛЕНИИ РАСПИСАНИЙ ЗАНЯТИЙ НА ОСНОВЕ ЭВОЛЮЦИОННЫХ МЕТОДОВ. АБУХАНИЯ АМЕР Ю.А.]. (это надо засунуть в список источников, а здесь заменить на номера [3,6] или т. п.)

(если не будет объема, можно сюда пару страниц по генетические алгоритмы запихнуть)

Локальная сложность описанной задачи состоит в большом наборе оцениваемых метрик, среди которых есть жесткие и слабые. Конкретно здесь требуется учитывать следующие направления:

1) Крайне не рекомендуется (а в случае с аудиториями и вообще запрещено) ставить занятия на слоты, в которых конкретные преподаватели не могут работать или аудитории уже заняты важными мероприятиями.

2) Очень желательно, чтобы аудитория соответствовала требованиям занятия, имела необходимое для этого вида занятий оборудование.

3) Группа (в виде всего потока, обычной группы или подгруппы) не может занимать аудитории, которые её не вмещают.

4) За неделю у каждой подгруппы должны быть проведены все занятия в нужном количестве (как минимум, это чуть менее обязательное условие в сравнении с предыдущими пунктами).

5) Требуется минимизировать общую сложность дней для каждой подгруппы, причём крайне не рекомендуется превышать некоторый порог сложности.

Так или иначе, описанные пункты отражают свои стороны «качества» требуемого расписания и должны быть оптимально скомбинированы так, чтобы искать решение, полезное в том же плане, в каком этого ждём мы. А это значит, что предстоит работа над поиском оптимальных гиперпараметров модели оптимизации.

Один из основных трюков при решении подобных задач генетическими алгоритмами – правильное представление решения, которое сократит объём вычислений и отбросит множество заведомо неподходящих решений (в нашем случае большие части пространства возможных решений вообще не будут соответствовать жёстким условиям). Здесь мы выбрали следующее представление решения. Решение описывается вектором, длина которого равна количеству слотов. Каждый элемент вектора – это допустимая комбинация четырёх компонент: преподаватель, предмет, аудитория, группа. Зная размеры групп, размеры аудиторий, информацию об оборудовании и пр., можно заранее сконструировать множество допустимых комбинаций, отчего любое конструируемое решение однозначно будет соответствовать жёстким условиям. Эти комбинации можно сохранить в таблице, а в векторе записывать номера строк из этой таблицы, сильно экономя память. Конечное решение мы будем получать в виде подобной таблицы (Таблица 1):

Таблица 1 – часть таблицы представления требуемого решения

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Преподаватель** | **Предмет** | **Аудитория** | **Группа** | **Слот** |
| Доц\_4 | Предмет\_11 | 7 | Группа\_1, Группа\_2 | 1 |
| Доц\_1 | Предмет\_3 | 2 | Группа\_2\_1 (2/1) | 2 |
| Доц\_3 | Предмет\_12 | 7 | Группа\_2 | 3 |
| Acc\_3 | Предмет\_5 | 13 | Группа\_1 | 4 |
| Проф\_2 | Предмет\_9 | 6 | Группа\_2\_2 (2/2) | 6 |
| Проф\_3 | Предмет\_2 | 7 | Группа\_2\_2 (2/2) | 7 |
| Проф\_1 | Предмет\_1 | 7 | Группа\_2 | 8 |
| Доц\_1 | Предмет\_3 | 17 | Группа\_2\_1 (2/1) | 9 |
| Асс\_2 | Предмет\_5 | 7 | Группа\_1 | 10 |
| Доц\_4 | Предмет\_11 | 20 | Группа\_2 | 11 |
| Доц\_4 | Предмет\_6 | 4 | Группа\_1, Группа\_2 | 12 |
| Доц\_3 | Предмет\_12 | 7 | Группа\_1 | 13 |
| Проф\_1 | Предмет\_3 | 20 | Группа\_2\_2 (2/2) | 14 |
| Доц\_2 | Предмет\_8 | 5 | Группа\_1\_1 (1/1) | 15 |
| Доц\_3 | Предмет\_12 | 4 | Группа\_1, Группа\_2 | 16 |

Здесь содержится вся необходимая информация, такую таблицу легко преобразовать в вид «преподаватель-группа», «группа-аудитория» и т. п.

Итак, мы представили решение в виде вектора длины количества слотов, где каждый элемент – номер допустимой комбинации преподавателя, аудитории, предмета и группы в сохранённой таблице. Поскольку элементы вектора – лишь номера строк, это дискретные значения, не имеющие какого-то присущего им порядка или расстояния (близости), поэтому в рамках генетического алгоритма никакая мутация невозможна, так как мутация немного «сдвигает» решение в пространстве, рассчитывая получить решение получше, но у нас нет непрерывного пространства. Единственная возможная мутация – это замена некоторых элементов на случайные, однако это не эффективно. Вместо мутации мы будем использовать следующий ход: каждые несколько поколений (скажем, после 25 поколений стагнации) делать восхождение по холму для нескольких самых лучших решений в популяции (скажем, для 50). Восхождение по холму – это, по сути, покоординатный спуск по нескольким случайным координатам.

Далее перечислены основные тонкости при реализации нашего оптимизационного алгоритма для решения задачи составления расписания:

1) Каждый элемент вектора решений соответствует конкретному слоту

2) Каждый элемент – номер комбинации в таблице допустимых комбинаций. При этом для каждого слота можно учесть требования преподавателей и аудиторий и создать своё множество допустимых значений. Кроме того, здесь же можно добавить фиктивные элементы, которые будут означать, что во время конкретного слота никакое занятие не проводится

3) Не используем мутацию, но делаем несколько итераций восхождения по холму для самых лучших представителей популяции после некоторого времени стагнации.

4) Для начальной популяции выбираем 10% лучших решений из большого случайного множества

5) Используем равномерное скрещивание и линейное ранжирование для селекции, а также элитизм

6) Позволяем алгоритму работать очень большое число поколений, но останавливаем после 400-500 поколений стагнации.

7) После завершения работы генетического алгоритма для нескольких лучших элементов в популяции производим максимально допустимое в рамках одного измерения восхождение по холму. Иными словами, для случайного элемента вектора смотрим, какое из всех возможных значений этого элемента даст минимум функции стоимости; заменяем на это значение, переходим к следующему случайному аргументу; выполняем, пока не окажется, что были проверены все элементы, но прогресса не случилось.

8) В случае прогресса в предыдущем пункте убираем из популяции все дубликаты и повторяем тот же генетический алгоритм, но без восхождения по холму и со стратегией первого самца. Иными словами, одним из родителей каждого следующего потомка будет самый лучший элемент в популяции.

Преимущества указанного алгоритма:

1) Быстрый выход на хорошее решение

2) Относительно небольшое (несколько минут) время работы на одном процессоре

3) Очень хорошая масштабируемость на любое число процессоров. Компания OpenAI в статье «Evolution Strategies as a Scalable Alternative to Reinforcement Learning» показала, как с помощью эволюционных алгоритмов и хорошего распараллеливания можно находить решения задач обучения с подкреплением всего на 10-20 минут!

Что можно сделать лучше:

1) Попытаться использовать разные стратегии оппозиционного обучения, в том числе революции

2) Использовать намного больше поколений и больший размер популяции (сейчас в популяции 200 особей)

3) Оптимизировать вычисления функции, тем самым сильно ускорив вычисления

4) Раз в несколько поколений разбавлять процесс оптимизации другими более трудно реализуемыми алгоритмами, позволяющими улучшить решение в конкретной ситуации

Рассмотрим простой пример работы алгоритма и весь процесс поиска решения. Программная реализация была написана на языке Python версии 3.8.5, дополнительно активно использовались пакеты:

1) geneticalgorithm2 для применения генетических алгоритмов; этот пакет прост в использовании, хорошо оптимизирован и обладает существенным функционалом для настройки собственных обратных вызовов и т. п.

2) DiscreteHillClimbing для восхождения по холму специально для дискретных задач

Итак, программе на вход необходимо предоставить Excel-файл с нужным таблицами. Среди них:

1) Таблица отображения таймслотов в реальные занятия в конкретные день и время (Таблица 2). Она нужна в основном для создания таблиц расписания занятий у группы, преподавателя и пр. Требуется отметить, что здесь используется упрощенный случай, поэтому исключена возможность проводить несколько занятий у разных групп в рамках одного таймслота, то есть параллельно.

Таблица 2 – расписание таймслотов

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Таймслоты | | | | | | |
| № | Пон | Вт | Ср | Чт | Пт | Сб |
| 1 | 1 | 8 | 15 | 22 | 29 | 36 |
| 2 | 2 | 9 | 16 | 23 | 30 | 37 |
| 3 | 3 | 10 | 17 | 24 | 31 | 38 |
| 4 | 4 | 11 | 18 | 25 | 32 | 39 |
| 5 | 5 | 12 | 19 | 26 | 33 | 40 |
| 6 | 6 | 13 | 20 | 27 | 34 | 41 |
| 7 | 7 | 14 | 21 | 28 | 35 | 42 |

2) Таблицы запретов у преподавателей/групп/аудиторий. В таблице для преподавателей (Таблица 3), к примеру, для каждого преподавателя отмечается, в каких таймслотах он не может работать.

Таблица 3 – запреты у преподавателей

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | Таймслоты |  |  |  |  |
| Преподаватель | 1 | 2 | 3 | 4 | 5 |
| Проф\_1 |  |  |  |  |  |
| Проф\_2 |  |  | 0 |  |  |
| Проф\_3 | 0 |  |  |  |  |
| Проф\_4 |  |  |  |  |  |
| Доц\_1 |  |  | 0 |  |  |
| Доц\_2 |  |  |  |  |  |
| Доц\_3 |  |  |  |  | 0 |
| Доц\_4 |  |  |  |  |  |
| Асс\_1 |  |  |  |  |  |
| Асс\_2 |  | 0 |  |  |  |
| Acc\_3 |  |  |  |  |  |

3) Таблица, определяющая, какие предметы может вести каждый преподаватель (Таблица 4).

Таблица 4 – предметы каждого преподавателя

|  |  |
| --- | --- |
| Имя | Дисциплины |
| Проф\_1 | Предмет\_1; Предмет\_3; Предмет\_7; |
| Проф\_2 | Предмет\_2; Предмет\_4; Предмет\_9; |
| Проф\_3 | Предмет\_5; Предмет\_2; |
| Проф\_4 | Предмет\_6; Предмет\_4; |
| Доц\_1 | Предмет\_7; Предмет\_3; |
| Доц\_2 | Предмет\_8; Предмет\_3; Предмет\_2; |
| Доц\_3 | Предмет\_2; Предмет\_12; |
| Доц\_4 | Предмет\_6; Предмет\_11; |
| Асс\_1 | Предмет\_4; Предмет\_7; Предмет\_4; |
| Асс\_2 | Предмет\_5; Предмет\_10; Предмет\_6; |
| Acc\_3 | Предмет\_3; Предмет\_5; |

4) Таблица аудиторий (Таблица 5), хранящая информацию об их вместимости и доступном оборудовании.

Таблица 5 – таблица аудиторий

|  |  |  |
| --- | --- | --- |
| Номер | Вместимость | Оборудование |
| 1 | 25 | Проектор |
| 2 | 85 |  |
| 3 | 30 | Проектор |
| 4 | 100 | ЭВМ |
| 5 | 20 | Проектор |
| 6 | 50 |  |
| 7 | 100 | Проектор, ЭВМ |
| 8 | 25 |  |
| 9 | 30 |  |
| 10 | 15 | ЭВМ |
| 11 | 15 | ЭВМ |
| 12 | 30 |  |
| 13 | 35 | ЭВМ, Проектор |
| 14 | 30 | Проектор |
| 15 | 33 | ЭВМ |
| 16 | 18 | ЭВМ |
| 17 | 24 |  |
| 18 | 23 | ЭВМ |
| 19 | 33 | Проектор |
| 20 | 50 | ЭВМ, Проектор |

5) Таблица предметов (Таблица 6), хранящая информацию о требуемом оборудовании, сложности и общем числе занятий.

Таблица 6 – таблица предметов

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Название | Тип | Требования к аудитории | Сложность | Кол-во занятий в неделю |
| Предмет\_1 | Лекция | Проектор | 3 | 2 |
| Предмет\_2 | Лекция | Проектор | 3 | 1 |
| Предмет\_3 | Лекция |  | 5 | 2 |
| Предмет\_4 | Практика | ЭВМ | 7 | 3 |
| Предмет\_5 | Практика | ЭВМ | 6 | 3 |
| Предмет\_6 | Практика | ЭВМ | 8 | 2 |
| Предмет\_7 | Лекция |  | 5 | 1 |
| Предмет\_8 | Семинар |  | 4 | 1 |
| Предмет\_9 | Семинар |  | 2 | 1 |
| Предмет\_10 | Лекция |  | 1 | 1 |
| Предмет\_11 | Лекция |  | 5 | 3 |
| Предмет\_12 | Практика | ЭВМ | 7 | 2 |

6) Таблица групп (Таблица 7), хранящая информацию об их объёме и отношениях, какая группа какую содержит. Эти отношения нужны для ускорения вычислений, чтобы информация от больших групп распространялась на все малые, которые в них содержатся.

Таблица 7 – таблица групп

|  |  |  |  |
| --- | --- | --- | --- |
| Код группы | Название | Содержит | Кол-во студ |
| 1 | Группа\_1, Группа\_2 | 2 3 | 72 |
| 2 | Группа\_1 | 4 5 | 33 |
| 3 | Группа\_2 | 6 7 | 39 |
| 4 | Группа\_1\_1 (1/1) |  | 16 |
| 5 | Группа\_1\_2 (1/2) |  | 17 |
| 6 | Группа\_2\_1 (2/1) |  | 19 |
| 7 | Группа\_2\_2 (2/2) |  | 20 |

Кратко продемонстрируем шаги алгоритма (для этого был создан специальный Jupyter-блокнот):

1) Импорт созданных модулей (Рис. 1)

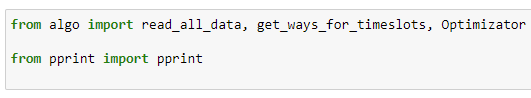


Рисунок 1 – импорт модулей

2) Преобразование Excel-файла в словарь полезных объектов (Рис. 2)

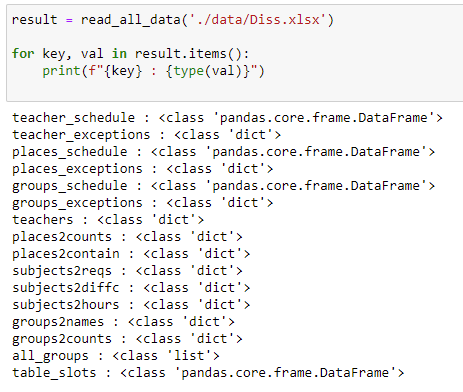


Рисунок 2 – преобразование входных данных

3) Создание таблицы допустимых комбинаций и словаря (Рис. 3), в котором для каждого тайм-слота содержатся индексы разрешенных комбинаций из созданной таблицы

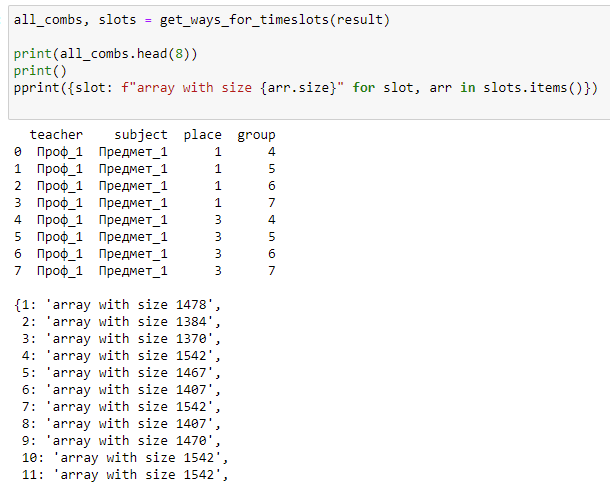


Рисунок 3 – создание допустимых комбинаций

4) Создание объекта алгоритма и инициализация метрик и другой требуемой для оптимизации информации (Рис. 4)

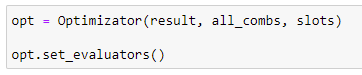


Рисунок 4 – подготовка алгоритма

5) Оптимизация описанным способом (Рис. 5-9), некоторые графики приведены отдельно,

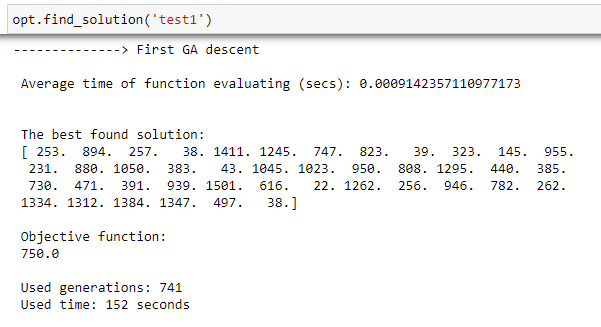


Рисунок 5 – результат оптимизации генетическим алгоритмом

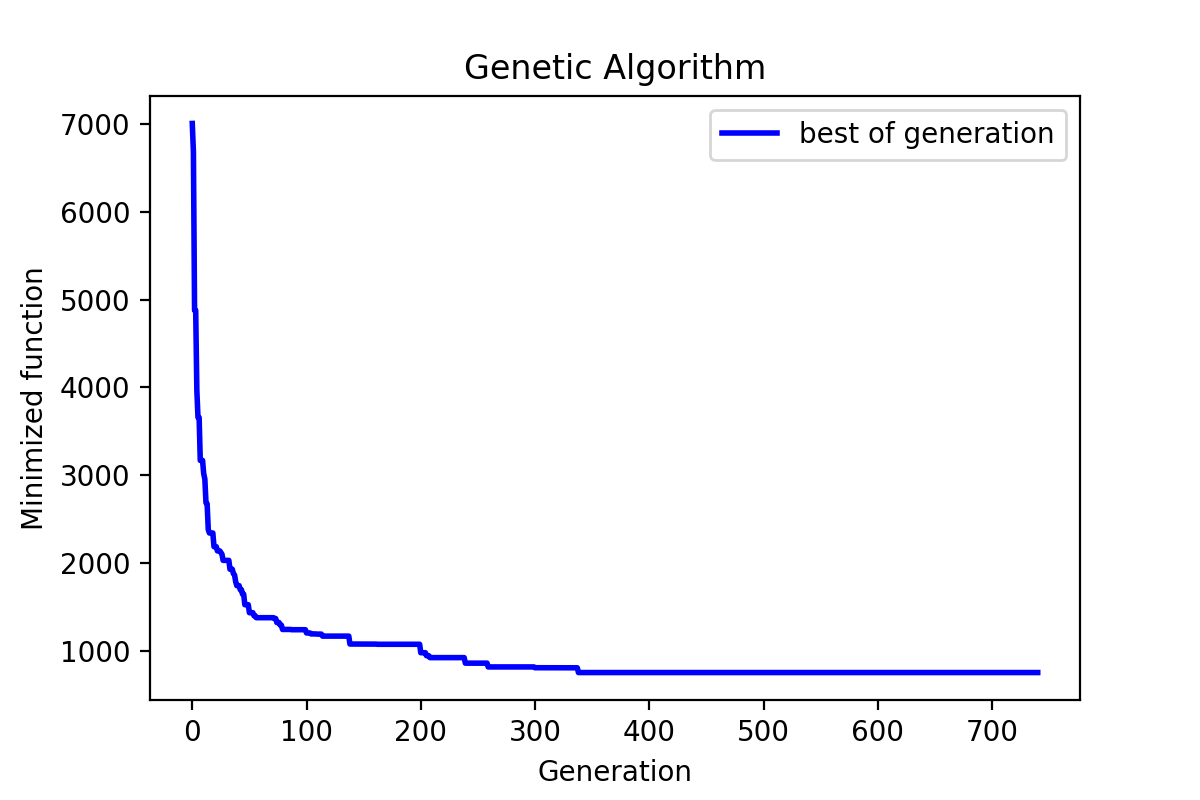


Рисунок 6 – изменение функции стоимости

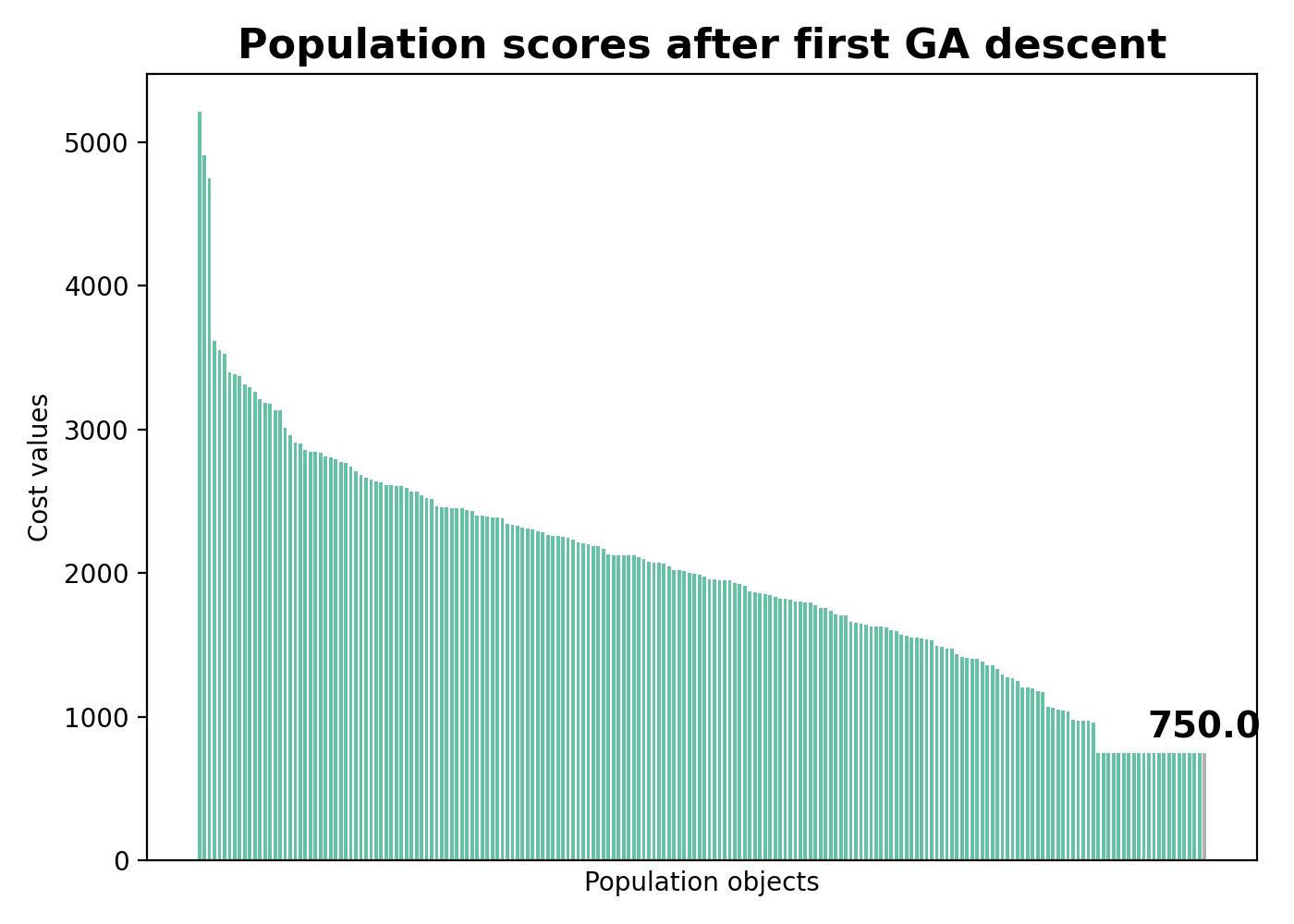


Рисунок 7 – стоимость для каждой особи в поколении в конце работы генетического алгоритма

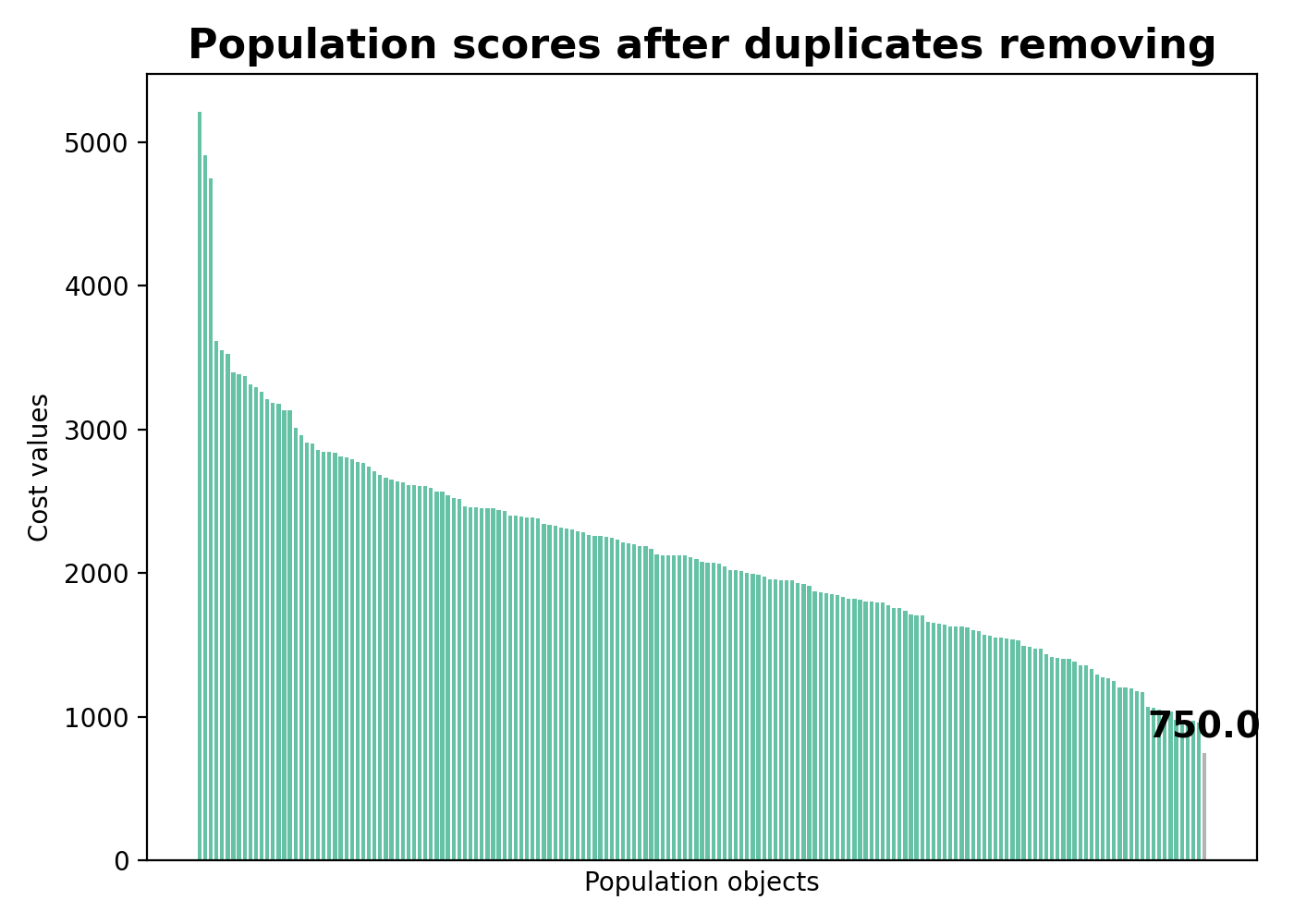


Рисунок 8 – поколение после удаления дубликатов

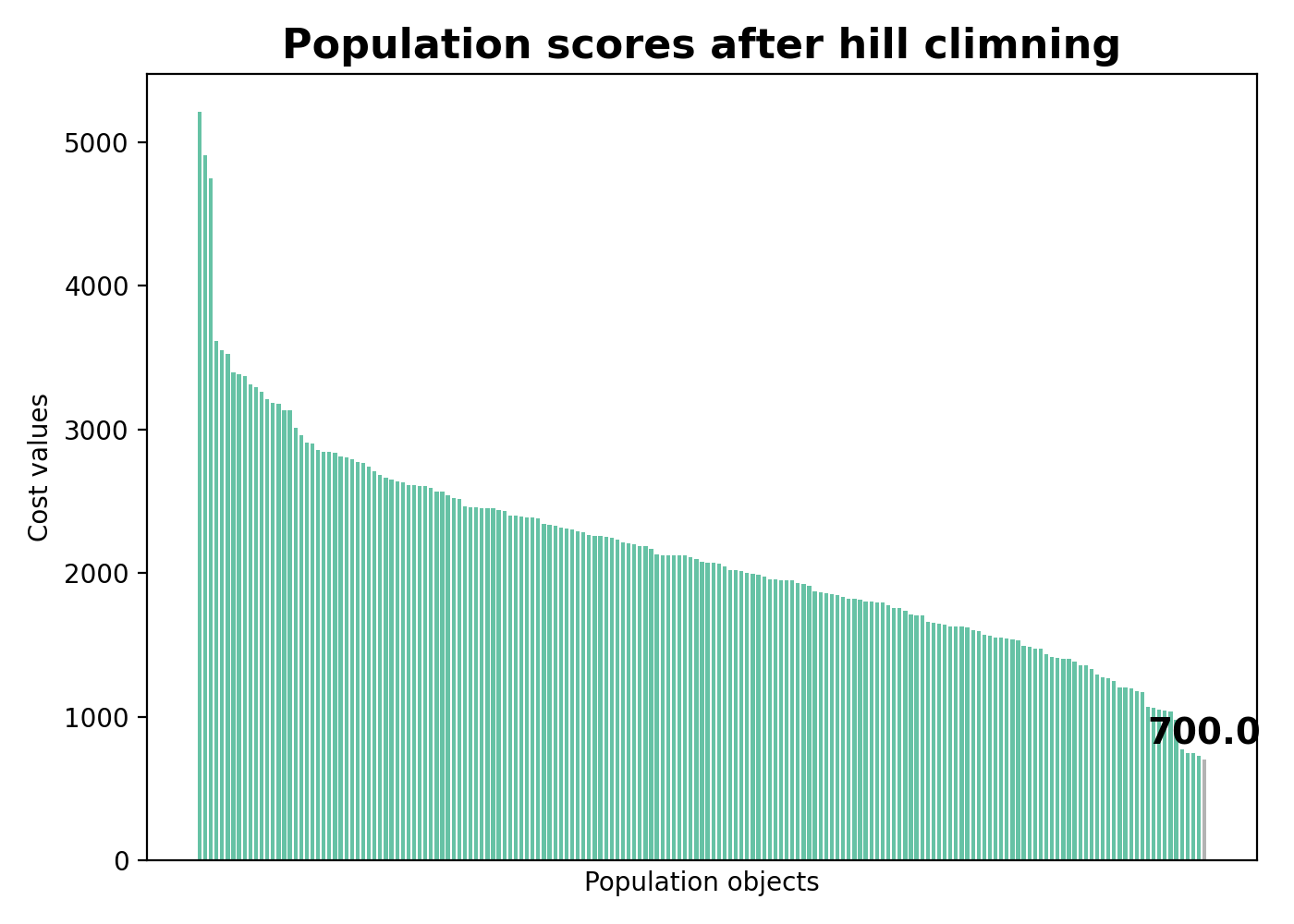


Рисунок 9 – поколение после восхождения по холму для лучших особей

Как можно заметить, первая оптимизация генетическим алгоритмом достигла метрики 750, а восхождению по холму удалось улучшить эту метрику до 700, что очень даже неплохо. К сожалению, вторая оптимизация генетическим алгоритмом не сделала улучшения на этот раз, но в случае неудовлетворённости результатом можно попробовать перезапустить все заново.

6) Преобразование и сохранение результатов командой opt.save\_results()

В результате мы получаем Excel-файл с найденным решением, некоторой справочной информацией и полезными таблицами, все расположены в своих вкладках. Основная таблица имеет вид (Таблица 8):

Таблица 8 – конечный вид найденного решения

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Преподаватель** | **Предмет** | **Аудитория** | **Группа** | **Слот** |
| Доц\_2 | Предмет\_8 | 15 | Группа\_1 | 1 |
| Доц\_4 | Предмет\_11 | 4 | Группа\_2 | 2 |
| Проф\_4 | Предмет\_6 | 4 | Группа\_1, Группа\_2 | 3 |
| Доц\_2 | Предмет\_2 | 7 | Группа\_1 | 4 |
| Проф\_1 | Предмет\_1 | 19 | Группа\_2\_2 (2/2) | 5 |
| Доц\_4 | Предмет\_11 | 2 | Группа\_1 | 6 |
| Проф\_1 | Предмет\_1 | 19 | Группа\_2\_1 (2/1) | 7 |
| Доц\_2 | Предмет\_8 | 5 | Группа\_2\_2 (2/2) | 8 |
| Acc\_3 | Предмет\_5 | 7 | Группа\_1, Группа\_2 | 9 |
| Проф\_1 | Предмет\_3 | 19 | Группа\_1 | 10 |
| Асс\_1 | Предмет\_4 | 7 | Группа\_1, Группа\_2 | 11 |
| Проф\_2 | Предмет\_9 | 14 | Группа\_1\_2 (1/2) | 12 |
| Проф\_2 | Предмет\_9 | 17 | Группа\_2\_1 (2/1) | 13 |
| Доц\_1 | Предмет\_3 | 4 | Группа\_2 | 14 |
| Проф\_4 | Предмет\_4 | 4 | Группа\_2 | 15 |
| Проф\_1 | Предмет\_1 | 13 | Группа\_1 | 16 |
| Асс\_2 | Предмет\_10 | 19 | Группа\_1 | 17 |
| Проф\_4 | Предмет\_4 | 4 | Группа\_2 | 18 |
| Доц\_4 | Предмет\_11 | 20 | Группа\_1 | 19 |
| Доц\_4 | Предмет\_11 | 4 | Группа\_2 | 20 |
| Acc\_3 | Предмет\_3 | 4 | Группа\_1 | 21 |
| Асс\_1 | Предмет\_4 | 20 | Группа\_1 | 22 |
| Доц\_3 | Предмет\_12 | 4 | Группа\_1, Группа\_2 | 23 |
| Проф\_2 | Предмет\_9 | 2 | Группа\_1\_1 (1/1) | 24 |
| Проф\_3 | Предмет\_2 | 3 | Группа\_2\_1 (2/1) | 25 |
| Доц\_1 | Предмет\_3 | 6 | Группа\_2 | 26 |
| Проф\_2 | Предмет\_2 | 1 | Группа\_2\_2 (2/2) | 27 |
| Проф\_3 | Предмет\_5 | 4 | Группа\_1, Группа\_2 | 28 |
| Доц\_2 | Предмет\_8 | 1 | Группа\_2\_1 (2/1) | 29 |
| Доц\_4 | Предмет\_6 | 7 | Группа\_2 | 30 |
| Проф\_1 | Предмет\_7 | 15 | Группа\_1 | 31 |
| Проф\_4 | Предмет\_4 | 7 | Группа\_1 | 32 |
| Проф\_2 | Предмет\_9 | 18 | Группа\_2\_2 (2/2) | 33 |
| Проф\_1 | Предмет\_1 | 20 | Группа\_1 | 34 |
| Асс\_2 | Предмет\_5 | 4 | Группа\_1, Группа\_2 | 35 |
| Асс\_1 | Предмет\_7 | 20 | Группа\_2 | 36 |
| Доц\_4 | Предмет\_6 | 13 | Группа\_1 | 37 |
| Асс\_2 | Предмет\_10 | 4 | Группа\_2 | 38 |
| Доц\_3 | Предмет\_12 | 7 | Группа\_1, Группа\_2 | 39 |
| Проф\_1 | Предмет\_1 | 1 | Группа\_2\_2 (2/2) | 40 |
| Доц\_4 | Предмет\_11 | 2 | Группа\_1, Группа\_2 | 41 |
| Проф\_1 | Предмет\_1 | 14 | Группа\_2\_1 (2/1) | 42 |

Также мы можем посмотреть на расписание преподавателей и аналогичные таблицы (Таблица 9).

Таблица 9 – расписание преподавателей

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **№** | **Пон** | **Вт** | **Ср** | **Чт** | **Пт** | **Сб** |
| 1 | Доц\_2 | Доц\_2 | Проф\_4 | Асс\_1 | Доц\_2 | Асс\_1 |
| 2 | Доц\_4 | Acc\_3 | Проф\_1 | Доц\_3 | Доц\_4 | Доц\_4 |
| 3 | Проф\_4 | Проф\_1 | Асс\_2 | Проф\_2 | Проф\_1 | Асс\_2 |
| 4 | Доц\_2 | Асс\_1 | Проф\_4 | Проф\_3 | Проф\_4 | Доц\_3 |
| 5 | Проф\_1 | Проф\_2 | Доц\_4 | Доц\_1 | Проф\_2 | Проф\_1 |
| 6 | Доц\_4 | Проф\_2 | Доц\_4 | Проф\_2 | Проф\_1 | Доц\_4 |
| 7 | Проф\_1 | Доц\_1 | Acc\_3 | Проф\_3 | Асс\_2 | Проф\_1 |

В таблице 10 приведена сложность для всех комбинаций подгрупп и дней недели.

Таблица 10 – итоговые сложности

|  |  |  |  |
| --- | --- | --- | --- |
| **Группа** | **День** | **Сложность** | **Штраф** |
| Группа\_1\_1 (1/1) | 1 | 20 | 20 |
| Группа\_1\_1 (1/1) | 2 | 18 | 18 |
| Группа\_1\_1 (1/1) | 3 | 14 | 14 |
| Группа\_1\_1 (1/1) | 4 | 22 | 42 |
| Группа\_1\_1 (1/1) | 5 | 21 | 31 |
| Группа\_1\_1 (1/1) | 6 | 20 | 20 |
| Группа\_1\_2 (1/2) | 1 | 20 | 20 |
| Группа\_1\_2 (1/2) | 2 | 20 | 20 |
| Группа\_1\_2 (1/2) | 3 | 14 | 14 |
| Группа\_1\_2 (1/2) | 4 | 20 | 20 |
| Группа\_1\_2 (1/2) | 5 | 21 | 31 |
| Группа\_1\_2 (1/2) | 6 | 20 | 20 |
| Группа\_2\_1 (2/1) | 1 | 16 | 16 |
| Группа\_2\_1 (2/1) | 2 | 20 | 20 |
| Группа\_2\_1 (2/1) | 3 | 19 | 19 |
| Группа\_2\_1 (2/1) | 4 | 21 | 31 |
| Группа\_2\_1 (2/1) | 5 | 18 | 18 |
| Группа\_2\_1 (2/1) | 6 | 21 | 31 |
| Группа\_2\_2 (2/2) | 1 | 16 | 16 |
| Группа\_2\_2 (2/2) | 2 | 22 | 42 |
| Группа\_2\_2 (2/2) | 3 | 19 | 19 |
| Группа\_2\_2 (2/2) | 4 | 21 | 31 |
| Группа\_2\_2 (2/2) | 5 | 16 | 16 |
| Группа\_2\_2 (2/2) | 6 | 21 | 31 |

Все жесткие требования выполнились. Все предметы отрабатываются необходимое количество занятий, по сложностям почти нет превышений отметки 20.