

|  |
| --- |
| МИНОБРНАУКИ РОССИИ |
| Федеральное государственное бюджетное образовательное учреждение  высшего образования  **"МИРЭА - Российский технологический университет"**  **РТУ МИРЭА** |

**Институт** Информационных Технологий

**Кафедра** Вычислительной Техники

**ПРАКТИЧЕСКАЯ РАБОТА №1**

**по дисциплине**

**«Теория принятия решений»**

**Метод Парето**

Студент группы: ИКБО-04-21 \_\_Даурбеков М.И.\_\_ *(Ф. И.О. студента)*

Преподаватель \_\_Железняк Л.М.\_\_

*(Ф.И.О. преподавателя)*

|  |  |  |
| --- | --- | --- |
|  |  |  |

Москва 2023

**СОДЕРЖАНИЕ**

[ВВЕДЕНИЕ 3](#_Toc130762925)

[1 МЕТОД ПАРЕТО 4](#_Toc130762926)

[1.1 Выбор Парето-оптимального множества 4](#_Toc130762927)

[1.2 Указание верхних/нижних границ критериев. 4](#_Toc130762928)

[1.3 Субоптимизация 4](#_Toc130762929)

[1.4 Лексикографическая оптимизация 4](#_Toc130762930)

[1.5 Результаты работы программы 4](#_Toc130762931)

[ЗАКЛЮЧЕНИЕ 6](#_Toc130762932)

[СПИСОК ИНФОРМАЦИОННЫХ ИСТОЧНИКОВ 7](#_Toc130762933)

[ПРИЛОЖЕНИЯ 8](#_Toc130762934)

ВВЕДЕНИЕ

Оптимизация по Парето использует отношение Парето-доминирования, которое отдаёт предпочтение одному объекту перед другим только» том случае, когда первый объект по всем критериям не хуже второго и хотя бы но одному из них лучше. При истинности этого условия первый объект считается доминирующим, а второй - доминируемым. Два объекта, для которых предпочтение хотя бы, по одному критерию расходится, считаются несравнимыми.

Для выбора одной оптимальной стратегии из множества эффективных решений в каждой конкретной многокритериальной задаче необходимо использовать дополнительную информацию.

1. МЕТОД ПАРЕТО

Состояние А (множество параметров) называется Парето-оптимальным, если не существует другого состояния В (множества других параметров), доминирующего состояние А относительно целевой функции. Состояние А доминирует состояние В, если хотя бы по одному параметру А лучше В, а по остальным не хуже.

Применительно к задаче переговоров этот принцип утверждает, что, если для ситуации В существует такая ситуация А, что выигрыш каждого из участников переговоров при реализации ситуации А не меньше, чем при реализации ситуации В и, по крайней мере, один переговорщик получит выигрыш строго больший, то они предпочтут ситуацию А ситуации В.

Если относительно пары альтернатив-решений одной и той же многокритериальной задачи нельзя сказать, какая из них лучше, то их называют несравнимыми. Множество таких альтернатив называются множеством Парето.

## **Выбор Парето-оптимального множества**

Необходимо принять решение о выборе высшего учебного заведения используя Метод Парето. В качестве критериев будут выступать: Цена обучения за один учебный год, Рейтинг ВУЗа, Минимальная сумма баллов за ЕГЭ (необходимая для поступления), Среднее время поездки от дома до ВУЗа, Количество бюджетных мест в прошлом году. В списке альтернатив участвуют: РАНХиГС, РТУ МИРЭА, МГТУ им. Баумана, МГУ, МТУСИ, МАИ, МЭИ, Московский Политех, Синергия. Стремления указаны в таблице 1.

Таблица 1. - Список альтернатив, критериев и стремлений.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| № | Варианты  Решений | Критерии | | | | |
| Цена  (-) | Рейтинг (+) | Мин. Баллы  (-) | Время поездки  (-) | Кол-во бюджетных мест  (+) |
| 1 | РАНХиГС | 250000 | 4.4 | 144 | 35 | 50 |
| 2 | РТУ МИРЭА | 200000 | 4.9 | 100 | 20 | 100 |
| 3 | МГТУ | 270000 | 4.6 | 115 | 120 | 70 |
| 4 | МГУ | 226000 | 4.9 | 100 | 30 | 120 |
| 5 | МТУСИ | 230000 | 4.5 | 120 | 100 | 40 |
| 6 | МАИ | 210000 | 3.6 | 90 | 90 | 120 |
| 7 | МЭИ | 220000 | 4.1 | 130 | 95 | 90 |
| 8 | Политех | 180000 | 4.0 | 100 | 150 | 100 |
| 9 | Синергия | 200000 | 3.5 | 90 | 80 | 130 |

Таблица 2. Таблица сравнений

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
| 1 | x | x | x | x | x | x | x | x | x |
| 2 | A2 | x | x | x | x | x | x | x | x |
| 3 | н | н | x | x | x | x | x | x | x |
| 4 | A4 | н | A4 | x | x | x | x | x | x |
| 5 | н | н | н | н | x | x | x | x | x |
| 6 | н | н | н | н | н | x | x | x | x |
| 7 | н | н | н | н | н | н | x | x | x |
| 8 | н | н | н | н | н | н | н | x | x |
| 9 | н | н | н | н | н | н | н | н | x |

# Указание верхних/нижних границ критериев.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| № | Варианты  Решений | Критерии | | | | |
| Цена  (-) | Рейтинг (+) | Мин. Баллы  (-) | Время поездки (мин)  (-) | Кол-во бюджетных мест  (+) |
| 1 | РАНХиГС | 250000 | 4.4 | 144 | 35 | 50 |
| 2 | РТУ МИРЭА | 200000 | 4.9 | 100 | 20 | 100 |
| 4 | МГУ | 226000 | 4.9 | 100 | 30 | 120 |

Установим границы для критерий "Рейтинг" и "Время поездки". Рейтинг не должен быть ниже 3.9, а время поездки не более 60. Согласно данному условию таблица 1 трансформируется в таблцу 3.

Таблица 3. Список альтернатив с учетом границ

Варианты, удовлетворяющие ограничениям: {1, 2, 4}. Оптимальными из них являются: {2, 4}.

**1.3 Субоптимизация**

Субоптимизацию производят следующим образом: выделяют один из критериев, а по всем остальным критериям назначают нижние границы. Оптимальным при этом считается исход, максимизирующий выделенный критерий на множестве исходов, оценки которых по остальным критериям не ниже назначенных. Пусть в примере главным критерием выступает цена; ограничения: рейтиг не меньше 4.6, время поездки не больше 30 минут. Отбросим варианты, которые не удовлетворяют данным ограничениям и составим таблицу 4.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| № | Варианты  Решений | Критерии | | | | |
| Цена  (-) | Рейтинг (+) | Мин. Баллы  (-) | Время поездки (мин)  (-) | Кол-во бюджетных мест  (+) |
| 2 | РТУ МИРЭА | 200000 | 4.9 | 100 | 20 | 100 |
| 4 | МГУ | 226000 | 4.9 | 100 | 30 | 120 |

Таблица 4.

Остаются варианты 2 и 4. Цена ниже во втором варианте. Этот вариант и будет оптимальным.

**1.4 Лексикографическая оптимизация**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| № | Варианты  Решений | Критерии | | | | |
| Цена  (-) | Рейтинг (+) | Мин. Баллы  (-) | Время поездки (мин)  (-) | Кол-во бюджетных мест  (+) |
| 2 | РТУ МИРЭА | 200000 | 4.9 | 100 | 20 | 100 |

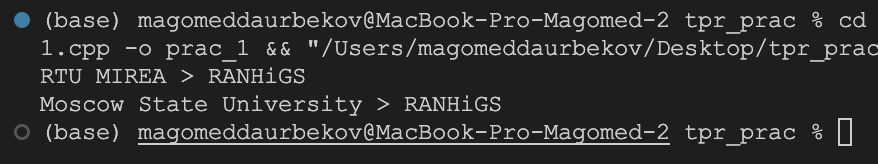
Лексикографическая оптимизация основана на упорядочении критериев по их относительной важности. После этого процедуру нахождения оптимального решения проводят следующим образом. На первом шаге отбирают исходы, которые имеют максимальную оценку по важнейшему критерию. Если такой исход единственный, то его и считают оптимальным. Если же таких исходов несколько, то среди них отбирают те, которые имеют максимальную оценку по следующему за важнейшим критерию. В результате такой процедуры всегда остается (по крайней мере, в случае конечного множества исходов) единственный исход — он и будет оптимальным.

Упорядочим критерии в примере по относительной важности, например, следующим образом: важнейший критерий – цена, следующий за ним по важности – время поездки. Отбросим варианты, которые не удовлетворяют данным ограничениям и составим таблицу 5.

Таблица 5.

Согласно таблице 5 остается только вариант 2. Он и будет оптимальным.

**1.5 Результаты работы программы**



**Рисунок 1 – Лексикографическая оптимизация.**

**ЗАКЛЮЧЕНИЕ**

Мы научились принимать решения методом Парето-оптимальности. Данный метод позволяет решать примитивные задачи принятий решений, однако в более сложных ситуациях метод будет считать все альтернативы несравнимыми.

**СПИСОК ИНФОРМАЦИОННЫХ ИСТОЧНИКОВ**

1. Болотова Л. С. Многокритериальная оптимизация. Болотова Л. С., Сорокин А. Б. [Электронный ресурс] / Метод. указания по вып. курсовой работы — М.: МИРЭА, 2015.
2. Сорокин А. Б. Методы оптимизации: гибридные генетические алгоритмы. Сорокин А. Б. [Электронный ресурс] / Учебно-метод. пособие — М.: МИРЭА, 2016.
3. Сорокин А. Б. Линейное программирование: практикум. Сорокин А. Б., Бражникова Е. В., Платонова О. В. [Электронный ресурс] / Учебно-метод. пособие — М.: МИРЭА, 2017.

**ПРИЛОЖЕНИЯ**

Приложение А – Код реализации метода Парето на языке C++.

**Приложение А**

Код реализации метода Парето на языке C++.

Листинг А.1. . Реализация Парето.

#include <iostream>

const char\* university\_names[] =

{

"RANHiGS",

"RTU MIREA",

"MGTU im. Baumana",

"Moscow State University",

"MTUSI",

"MAI",

"MEI",

"Politeh",

"Sinergia",

};

const float variants[][5] =

{

{250000, 4.4, 144, 35, 50},

{200000, 4.9, 100, 20, 100},

{270000, 4.6, 115, 120, 70},

{226000, 4.9, 100, 30, 120},

{230000, 4.5, 120, 100, 40},

{210000, 3.6, 90, 90, 120},

{220000, 4.1, 130, 95, 90},

{180000, 4.0, 100, 150, 100},

{200000, 3.5, 90, 80, 130},

};

const int num\_vars = sizeof(variants) / sizeof(\*variants);

const int num\_conds = sizeof(\*variants) / sizeof(float);

const float conditions[num\_conds] = { -1, 1, -1, -1, 1, };

const int priority[num\_conds] = { 5, 3, 1, 4, 2 };

const float rating\_min = 3.9;

const float time\_min = 60;

int main()

{

for(int i = 0; i < num\_vars; i++)

{

if(variants[i][1] < rating\_min)

continue;

if(variants[i][3] > time\_min)

continue;

int best\_prior = 0;

for(int j = i + 1; j < num\_vars; j++)

{

int num\_comps[3] = {0}; // 0 - лучше, 1 - равно, 2 - хуже

int num\_priors[5] = {0};

for(int k = 0; k < num\_conds; k++)

{

if(variants[j][k] \* conditions[k] > variants[i][k] \* conditions[k])

{

num\_comps[0]++;

num\_priors[priority[j] - 1]++;

}

else if(variants[j][k] \* conditions[k] == variants[i][k] \* conditions[k])

{

num\_comps[1]++;

num\_priors[priority[j] - 1]++;

}

else

num\_comps[2]++;

}

if(num\_comps[0] && num\_comps[0] + num\_comps[1] == num\_conds)

std::cout << university\_names[j] << " > " << university\_names[i] << std::endl;

}

}

return 0;

}