МОСКОВСКОЙ ТЕХНОЛОГИЧЕСКИЙ УНИВЕРСИТЕТ

Кафедра «Управление и моделирование систем»

Предмет «Формализованные модели и методы решения аналитических задач»

Домашняя работа №1 на тему «Методы Парето оптимизации. Сравнение методов.»

Ф.И.О. студента: Васильев А.В.
Группа БКБО-01-13
Шифр студента 130097
Курс 4
Подпись студента ____________

Ф.И.О. преподавателя Серов В.А.
Подпись преподавателя ____________
Дата__________

Москва, 2016 г.

ЛП последовательсть.

Постановку задачи выбора начального приближения представим в виде:

$$\text{определить } \underset{q \in Q}{C_{\Omega}^{D}}\left(I(Q)\right) \qquad (2.39)$$

То есть требуется определить множество $C_{\Omega}^{D}\left(I(q)\right)$ значений показателя $I(q)$, являющееся дискретной аппроксимацией ядра $C_{\Omega}\left(I(Q)\right)$ отношения предпочтения, выраженного в форме выпуклого полиэдрального конуса Ω, на множестве достижимых векторных оценок $I(Q)$.

Для решения задачи (2.39) предлагается использовать модифицированный алгоритм метода зондирования пространства параметров, основанный на методике ЛП $_{\tau}$-поиска [7] , и состоящий из двух основных этапов:

1) составление таблицы испытаний;

2) Ω-оптимизация таблицы испытаний.

Этап 1. Генерируется последовательность точек $\left\{p^{i}\right\}$, равномерно распределенная в r-мерном единичном гиперкубе Π_p. Наилучшими характеристиками равномерности распределения обладают так называемые ЛП $_{\tau}$-последовательности. Для генерации ЛП $_{\tau}$ -последовательности предлагается арифметический алгоритм, в котором используется специальная таблица направляющих чисел $\left\{r_{ij}\right\}$.

По заданному номеру i вычисляем

$$m = 1 + \left[ln(i) / ln(2)\right] \qquad (2.40)$$

Далее для $j = \overline{1,r}$ вычисляем

$$p_j^i = \sum_{k=1}^{m} 2^{(-k+1)} \left\{0.5 \times \sum_{l=k}^{m} \left[2\left\{i2^{(-l)}\right\}\right] \times \left[2\left\{r_{jl} 2^{(k-1-l)}\right\}\right]\right\}, \qquad (2.41)$$

где $[*]$, $\{*\}$ - целая и дробная части аргумента * соответственно.

После этого с помощью линейного преобразования $\boldsymbol{L}$, сохраняющего равномерность распределения, преобразуем множество сгенерированных точек $\left\{\boldsymbol{p}^i, \quad i=\overline{1, N_p}\right\} \in \boldsymbol{\Pi}_p$ в множество точек $\left\{\boldsymbol{q}^i, \quad i=\overline{1, N_p}\right\} \in \boldsymbol{\Pi}_q$, равномерно заполняющих r-мерный гиперпараллелепипед $\boldsymbol{\Pi}_q$, определяемый верхними и нижними ограничениями на параметры задачи $\boldsymbol{q}_L$ и $\boldsymbol{q}_H$:

$$\boldsymbol{\Pi}_q = \boldsymbol{L}\left(\boldsymbol{\Pi}_p\right). \tag{2.42}$$

Преобразование $\boldsymbol{L}$ задаем в виде

$$q_j^{(i)} = q_{Lj} + p_j^{(i)}\left(q_{Hj} - q_{Lj}\right) \tag{2.43}$$

Если в точке $\boldsymbol{q}^{(i)}$ выполняются линейные ограничения общего вида, входящие в описание множества $\boldsymbol{Q}$, то точка $\boldsymbol{q}^{(i)}$ является допустимой. В ней вычисляется $\boldsymbol{I}\left(\boldsymbol{q}^{(i)}\right)$ и заносится в таблицу испытаний $\boldsymbol{T}_q$.

Этап 2. Осуществляется оптимизация таблицы испытаний по конусу доминирования, заданному в виде (2.5). С этой целью из таблицы $\boldsymbol{T}_q$ выбирается какая-либо точка $\boldsymbol{q}^{(i)}$ и помечается. Просматривая все точки $\boldsymbol{q}^{(j)}$ таблицы $\boldsymbol{T}_q$, отличные от $\boldsymbol{q}^{(i)}$, исключим те из них, для которых

$$\boldsymbol{B}\left(\boldsymbol{I}\left(\boldsymbol{q}^{(j)}\right) - \boldsymbol{I}\left(\boldsymbol{q}^{(i)}\right)\right) \geqq \boldsymbol{0}_p, \tag{2.44}$$

т.е. проверяется условие

$$\left(\boldsymbol{I}\left(\boldsymbol{q}^{(j)}\right) - \boldsymbol{I}\left(\boldsymbol{q}^{(i)}\right)\right) \in -\boldsymbol{\Omega}.$$

Затем среди оставшихся точек выбирается ранее не помеченная, и вновь повторяется процесс исключения по правилу (2.44). После конечного числа шагов останутся только помеченные точки, принадлежащие множеству $\boldsymbol{C}_{\boldsymbol{\Omega}}^{\mathrm{D}}\left(\boldsymbol{I}(\boldsymbol{Q})\right)$.

Время оптимизации таблицы испытаний $\boldsymbol{T_q}$ уменьшается с сокращением интервалов неопределенности $\left[\boldsymbol{\lambda}_L, \ \boldsymbol{\lambda}_H\right]$. С увеличением плотности $ЛП_\tau$-сети имеет место

$$\boldsymbol{C}_{\boldsymbol{\Omega}}^{\mathrm{D}}\big(\boldsymbol{I}(\boldsymbol{Q})\big) \xrightarrow[N_p \to \infty]{} \boldsymbol{C}_{\boldsymbol{\Omega}}(\boldsymbol{I}(\boldsymbol{Q})).$$

Задача многокритериальной оптимизации

$$\text{Определить } \min(\max) \mathrm{F}(\mathrm{X}).$$
$$X \epsilon D$$

$X \epsilon E^n$ – n-мерный вектор варьируемых параметров;

$D \subset E^n$ – область допустимых значений(альтернатив) X;

$F(X) = [f_1(X), \dots, f_m(X)]^T \in E^m$ – векторная целевая функция.

Поиск оптимального решения $X^* \epsilon D$ сводится к выбору оптимальной векторной оценки $F(X^*) = F^*$ из множества достижимых векторных оценок

$$F(D) = \{ F(X) \in E^m | X \in D\},$$

Которая доставляет всем компонентам векторной целевой функции F(X) минимальные(максимальные) значения.

Бинарные отношения

Опр1. Бинарным отношением ϑ на множестве F(D) называется совокупность упорядоченных пар (F^1, F^2), где $F^1, F^2 \in F(D)$. Если $(F^1, F^2) \in \vartheta$, то говорят, что F^1 и F^2 находятся в отношении ϑ и пишут: $F^1 \vartheta F^2$

Опр2. Отношение строгого предпочтения.

$P: F^1 P\ F^2 <=> F^1 \leq F^2 <=> f_i^2 \leq f_i^2, i = \overline{1,m}$ и $F^1 \neq F^2$

$F^1 - F^2 \in \Omega = \{E^2 \leq \backslash O\}$

Опр3. Элемент F^* называется минимальным (недетерминированным) по Р на множестве F(D), если в F(D) не ∃ элемента $\tilde{F}$, для которого: $\tilde{F}PF^*$.
Опр4. Векторная оценка F^* минимальная по Р вида (3), называется оптимальной по Парето (эффективной). Множество векторных оценок, минимальных по Р, называется множеством Парето (эффективным множеством).

Алгоритм многокритериального ранжирования.

Шаг 1. Полагаем К=1;
Шаг 2. Вычислить β_k – количество точек, для которых справедливо

$$F^i - F^k \ \in \ \Omega \,, i = \overline{1, N}, i \neq k$$

Шаг 3. Вычислить функцию(индекс эффективности)

$$\Phi_k = \frac{1}{1 + \frac{\beta_k}{N-1}}$$

Шаг 4. Если $k < N$, полагаем $k = k + 1$. Переход к шагу 2. Иначе переход к шагу 5.
Шаг 5. Из точек множества $F(D)$ формируем подмножество $F_p(D) \subset F(D)$, обладающее свойством:

$$\forall F \in F(D) : \ \Phi(F) = 1$$

Это $F_p(D)$ будет множеством Парето.

Алгоритм вычеркивания конусами.

Шаг 1. Полагаем К=1;
Шаг 2. Удалить из дальнейшего рассмотрения точки, для которых справедливо

$$F^k - F^i \ \in \ \Omega \,, i = \overline{1, N}, i \neq k$$

Шаг 3. Если $k < N$, полагаем $k = k + 1$. Переход к шагу 2. Иначе переход к шагу 4.
Шаг 4. Получившиеся множество $F_p(D) \subset F(D)$.

Это $F_p(D)$ будет множеством Парето.

Свойства $\Phi(F)$

1. $\Phi_{max} = 1$, при $\beta_k = 0$
2. $\Phi_{min} = \frac{1}{2}$, при $\beta_k = N - 1$
3. $\frac{1}{2} \leq \Phi(F) \leq 1, \forall F \in \ F(D)$
4. $\Phi(F) = 1, \forall F \in F_p(D)$

Задание

Найти множество Парето двух функций с помощью алгоритмов индексов эффективности и вычеркивания конусом.

$$f(x) = 0.2(x_1^2) + 0.7(x_2^2) \rightarrow min$$

$$f(x) = 0.3(x_1 - 2) + 0.5((x_2 - 3)^2) \rightarrow min$$

Исходный код

```
%lutau.m
clear all
n=2;
```

```
q = [];
count=900;
table=zeros(2,count);
qq=zeros(2,count);
for i=1:count
   q = lptau(i,n);
   q(1) = q(1);
   q(2) = q(2);
   qq(1,i) = q(1);
   qq(2,i) = q(2);

   table(1,i) = f1(q(1),q(2));
   table(2,i) = f2(q(1),q(2));
end
table = sortrows(table');
tic%засекаем время работы алгоритма
table = indexF(table);
toc
for i=1:count%рисуем ЛП-сетку
    plot(qq(1,i),qq(2,i),'.','MarkerSize',5);
    hold on
end
figure
for i=1:count
    if(table(i,3)==1)%Парето точки, зеленый цвет
        plot(table(i,1),table(i,2),'.','MarkerSize',5,'color','green');
      end
      if(table(i,3)==0.9)%желтые цвет
        plot(table(i,1),table(i,2),'.','MarkerSize',5,'color','yellow');
      end
      if(table(i,3)==0.75)%оранжевый
        plot(table(i,1),table(i,2),'.','MarkerSize',5,'color',[255
,165,0]/255);
       end
      if(table(i,3)==0.5)%красный
        plot(table(i,1),table(i,2),'.','MarkerSize',5,'color','red');
       end
   hold on
end



%lptau.m генерация сетки
function f = lptau(i,n)
    NR = [
    1, 1, 1, 1, 1, 1, 1, 1, 1, 1;
    1, 3, 5, 15, 17, 51, 85, 255, 257, 771;
    1, 1, 7, 11, 13, 61, 67, 79, 465, 721;
    1, 3, 7, 5, 7, 43, 49, 147, 439, 1013;
    1, 1, 5, 3, 15, 51, 125, 141, 177, 759;
    1, 3, 1, 1, 9, 59, 25, 89, 321, 835;
    1, 1, 3, 7, 31, 47, 109, 173, 181, 949;
    1, 3, 3, 9, 9, 57, 43, 43, 225, 113;
    ];
 a=i;
 m=1+floor(log(a)/0.693147);
 q=zeros(n,1);
 for j=1:n
    s=0;
    for k=1:m
        ns=0;
        for l=k:m
           b = NR(j,l);
           ns = ns + floor(2*D(a/(2^l)))*floor(2*D(b/(2^(l+1-k))));
```

```
        end
        s = s + D(0.5*ns)/(2^(k-1));
    end
    q(j) = s;
 end
 f = q;
end
```

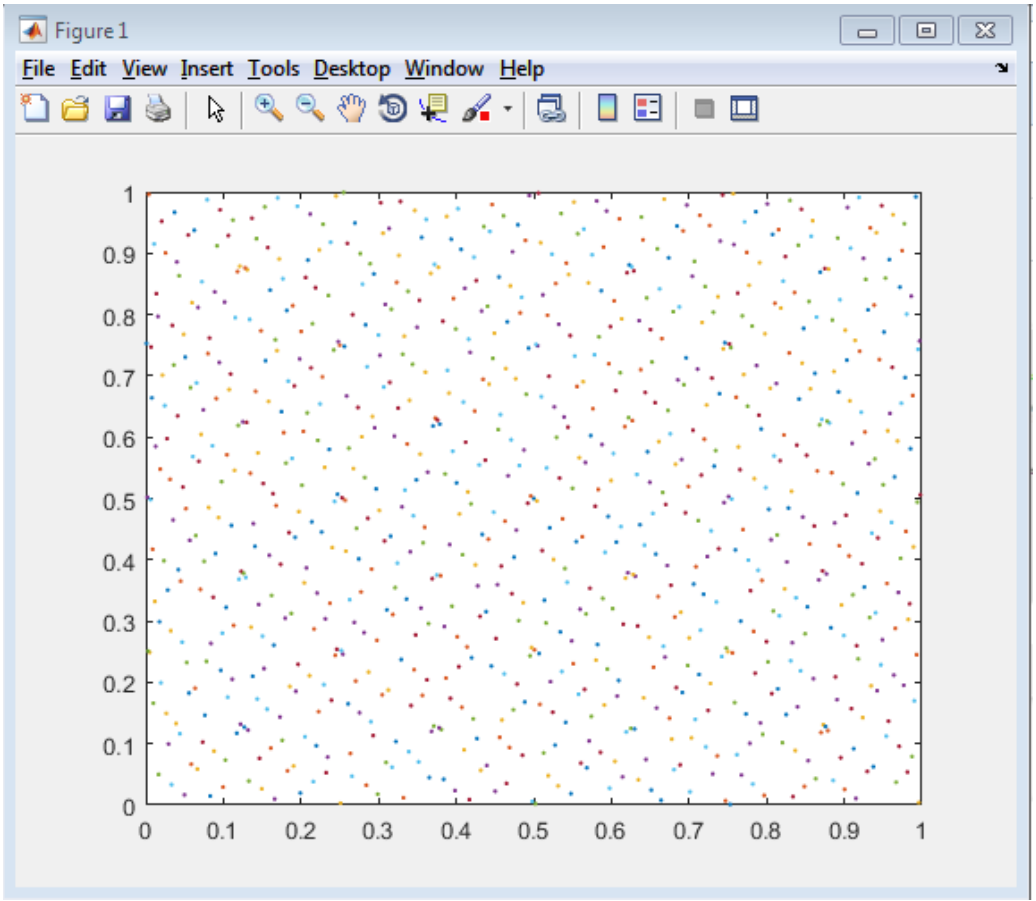
```
%indexF.m вычисляет индексы эффективность
function f = indexF(table)
 %вычисляем b(i)
 count = length(table);
b = zeros(1,count);
pareto = zeros(count,1);
k=0;
for i=1:count
    for j=1:count
        if(i~=j)
            if (((table(i,1)>table(j,1)) && (table(i,2)>table(j,2)) ))
                b(i) = b(i) + 1;
            end
        end
    end
     if (b(i)==0)%зеленые
            pareto(i)=1;
            table(i,3)=1;
            k=k+1;
     end
     if (b(i)>0 && b(i)<count*0.02)%желтые
         table(i,3)=0.9;
     end
     if (b(i)>=count*0.02 && b(i)<count*0.1)%оранжевые
         table(i,3)=0.75;
     end
     if (b(i)>=count*0.1)%красные
         table(i,3)=0.5;
     end
end
f=table;
end
```

```
%f1.m
function f= f1(x1,x2)
    f = 0.2*(x1^2)+0.7*(x2^2);
end
```

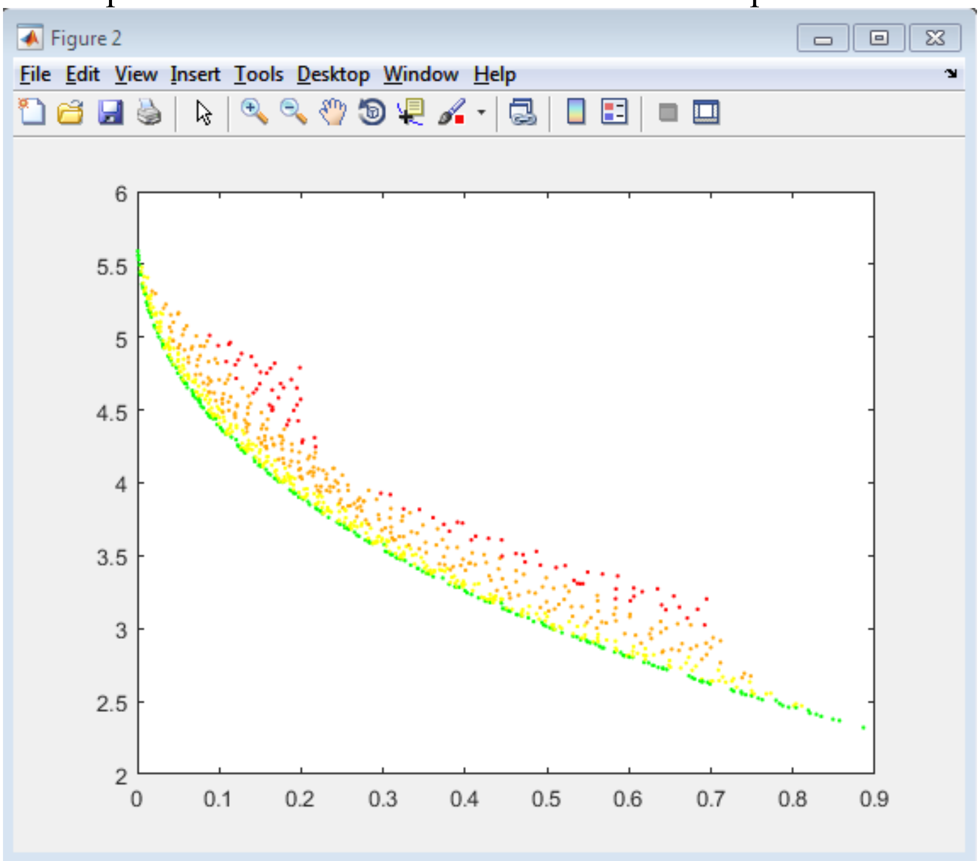
```
%f2.m
function f = f2(x1,x2)
    f = 0.3*((x1-2)^2)+0.5*((x2-3)^2);
end
```

Примеры работы

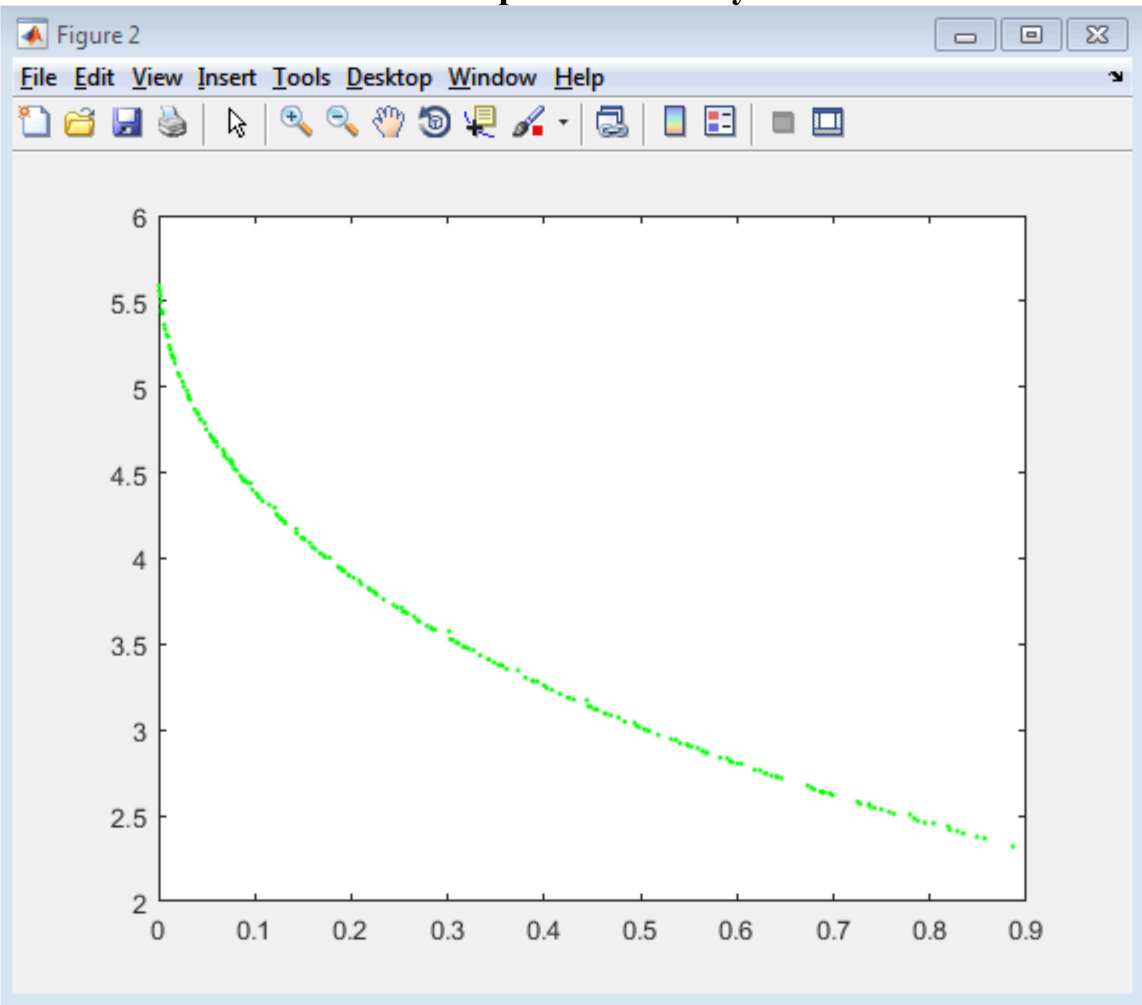
ЛП-сетка



Ранжирование. Зеленые точки – множество Парето.



Вычеркивание конусами.



Сравнение быстродействия

Ранжирование

```
>> lutau
Elapsed time is 0.130685 seconds.
```

Конусы

```
>> lutau
Elapsed time is 0.037374 seconds.
```