

## РЕШЕНИЯ В ДВУКРИТЕРИАЛЬНОЙ ЗАДАЧЕ ДРОБНО-ЛИНЕЙНОГО ПРОГРАММИРОВАНИЯ\*

Р.Г. Гамидов<sup>1</sup>, Н.К. Аллахвердиева<sup>1</sup>

<sup>1</sup>Бакинский Государственный Университет, Баку, Азербайджан

**Резюме.** В работе рассматривается двукритериальная задача дробно-линейного программирования. Предлагается способ построения паретовой границы (ПГ) и алгоритм поиска окончательного компромиссного решения на ПГ. Поиск осуществляется на основе дополнительной информации, полученной от лица принимающего решения (ЛПР). Далее, предлагается схема разложения, которая делает предложенный процесс принятия решения доступным для задач большой размерности, когда учитываются их специфики в применении в прикладных задачах. Даются графические иллюстрации хода выполнения предложенных процедур.

**Ключевые слова:** линейное программирование (ЛП), опорное решение, дробно-линейная функция, эффективное решение, симплекс-метод, большая задача, алгоритм разложения, Паретовая граница.

**AMS Subject Classification:** 90C31.

### 1. Введение

Роль двукритериальных задач математического программирования в многокритериальных задачах принятия решения особая. Методы исследования таких задач используются не только в многокритериальных задачах принятия решения, но также, например, при разработке процедур принятия решения в больших задачах и в игровых задачах [1,4]. Методы решения задачи ЛП (например, симплекс-метод) более часто применяются для анализа двукритериальных задач. С их применением разработаны различные алгоритмы построения ПГ [2,3-5]. Успех алгоритмов принятия решения во многом зависит насколько эффективно точно или приближенно представляется вся ПГ или нужная ее часть в процессе принятия решения.

В данной работе рассматриваем двукритериальную задачу математического программирования

$$F_1(x) \rightarrow \max, \quad F_2(x) \rightarrow \max, \quad x \in X \subset E^n,$$

где  $F_1$  и  $F_2$  - дробно-линейные функции,  $X$ - подмножество  $n$ -мерного Евклидова пространства  $E^n$ , означает множество допустимых решений. Оно формируется подобно, как в задачах ЛП.

Изучаем свойства ПГ. Показывается, что эта граница, в отличие от случая, когда

---

\* Работа была представлена на семинаре Института Прикладной Математики 08.12.2015

$F_1$  и  $F_2$  линейные, не является графиком вогнутой функции со свойством монотонности. Приводится числовой пример, подтверждающий этот факт. Однако, свойства монотонности и непрерывности сохраняются. Это помогает нам использовать возможности ЛП в процессе построения ПГ.

Решаемые в процессе построения ПГ задачи ЛП используют кроме ограничения  $x \in X$  и одно дополнительное ограничение. Если  $X$  обладает нужными свойствами, которые помогают применять более эффективные способы решения ЛП с ограничением  $x \in X$ , то приходится преодолевать трудности, связанные с присутствием дополнительного ограничения. В работе предлагается простая схема разложения, которая решает эту проблему.

## 2. Постановка задачи

Рассматриваемая задача формально представляется следующим образом:

$$F_1 = \frac{f_1}{g_1} = \frac{c^1 x + e^1}{d^1 x + f^1} \rightarrow \max, \quad F_2 = \frac{f_2}{g_2} = \frac{c^2 x + e^2}{d^2 x + f^2} \rightarrow \max, \\ Ax \leq b, \quad x \geq 0. \tag{1}$$

Здесь  $A - (m \times n)$  – матрица,  $x - n$  – мерный вектор столбец переменных  $x_i, i = 1, 2, \dots, n$ ,  $c^1, c^2, d^1, d^2$  – заданные  $n$  – мерные вектор-строки,  $b$  – заданный  $m$  – мерный вектор-столбец,  $e^1, e^2, f^1, f^2$  – заданные действительные числа.

Предполагается, что множество допустимых решений  $X$  ограничено и  $d^i x + f^i > 0, i = 1, 2$ , при  $x \in X$ . Прибавляя достаточно большое положительное число на  $F^i, i = 1, 2$ , можно обеспечить и условие  $c^i x + e^i > 0, i = 1, 2, x \in X$ . Тогда множество допустимых оценок

$$F(X) = \{(F_1(x), F_2(x)) \mid x \in X\}$$

будет компактным множеством в первой четверти координатной плоскости  $E^2$ .

Множество эффективных оценок (т.е. ПГ):

$$Y^P = \{y^0 \in F(X) \mid y \in E^2, y \neq y^0, y \geq y^0 \Rightarrow y \notin F(X)\}$$

тоже будет компактным множеством, которое состоит из части границы множества  $F(X)$ .

Рассматриваются следующие аспекты принятия решения задачи (1): 1) предлагается способ кусочно-линейной аппроксимации ПГ  $Y^P$ ; 2) приводится диалоговая форма организации поиска окончательного решения на ПГ; 3) изучается вопрос, как приспособить предложенную схему принятия

решения для специальных задач большой размерности. С этой целью разрабатывается алгоритм разложения для дробно-линейного программирования с одним дополнительным ограничением.

### 3. Кусочно-линейная аппроксимация $Y^P$

Рассмотрим индивидуальные задачи оптимизации

$$F_i(x) \rightarrow \max, \quad x \in X, \quad i = 1, 2$$

и пусть  $x^1, x^2$  их оптимальные решения, соответственно. Допустим, что решения  $x^1$  и  $x^2$  эффективные. Обозначим

$$M = (F_1(x^1), F_2(x^1)), N = (F_1(x^2), F_2(x^2)).$$

Тогда точки  $M$  и  $N$  будут крайними точками ПГ.

Обозначим  $\underline{\lambda} = F_1(x^2), \bar{\lambda} = F_1(x^1)$  и рассмотрим задачу

$$\frac{c^2x + e^2}{d^2x + f^2} \rightarrow \max, \quad \frac{c^1x + e^1}{d^1x + f^1} \geq \lambda, \quad x \in X, \quad \underline{\lambda} < \lambda < \bar{\lambda}. \quad (2)$$

Ее оптимальное решение обозначим через  $x(\lambda)$ . Тогда справедлива

**Лемма 1.**

$$\frac{c^1x(\lambda) + e^1}{d^1x(\lambda) + f^1} = \lambda. \quad (3)$$

**Доказательство.** Допустим, что (3) не имеет место, т. е.

$$\frac{c^1x(\lambda) + e^1}{d^1x(\lambda) + f^1} > \lambda.$$

Поскольку  $x^2$  эффективное решение

$$F_2(x(\lambda)) = \frac{c^2x(\lambda) + e^2}{d^2x(\lambda) + f^2} < \frac{c^2x^2 + e^2}{d^2x^2 + f^2} = F_2(x^2). \quad (4)$$

Задача (2) является задачей дробно-линейного программирования. Ее локальные экстремумы одновременно и являются глобальными. Тогда  $x(\lambda)$  будет оптимальным решением задачи  $F_2(x(\lambda)) \rightarrow \max, x \in X$ . Поскольку оно является ее локальным экстремумом, т.е.

$$F_2(x(\lambda)) = F_2(x^2).$$

А это противоречит (4). Лемма доказана.

Из леммы 1 следует, что отрезок  $[\underline{\lambda}, \bar{\lambda}]$  является проекцией ПГ на горизонтальную ось (ось абсцисс) координатной плоскости  $E^2$ , где находится ПГ. Далее, из этой леммы и компактности  $X$ , как следствие, следует справедливость утверждения следующей леммы.

**Лемма 2.** ПГ является графиком строго убывающей непрерывной функции от аргумента  $\lambda$ ,  $\underline{\lambda} \leq \lambda \leq \bar{\lambda}$ .

Лемма 2 позволяет кусочно-линейно аппроксимировать ПГ с любой заранее заданной точностью с помощью разбиения отрезка  $[\underline{\lambda}, \bar{\lambda}]$  по частям. Таким образом, задачу построения ПГ ( $Y^P$ ) свели к решению однопараметрической задачи дробно-линейного программирования:

$$F_2(x) \rightarrow \max, \quad F_1(x) = \lambda, \quad x \in X, \quad \underline{\lambda} \leq \lambda \leq \bar{\lambda}. \quad (5)$$

В случае, когда  $F_1$  и  $F_2$  линейные функции, для построения ПГ можно использовать методы решения параметрической задачи линейного программирования вида

$$F_1(x) + \lambda(F_2(x) - F_1(x)) \rightarrow \max, \quad x \in X, \quad 0 \leq \lambda \leq 1$$

Однако, в нашем случае, как показывает следующий пример, не всегда это возможно.

**Пример.** (который показывает не выпуклость и не вогнутость  $Y^P$ ).

$$F_1 = \frac{-x_2 + x_3 + 1}{x_1 - 0.75x_2 + 1} \rightarrow \max, \quad F_2 = \frac{3}{x_1 + x_2 + x_3 + 1} \rightarrow \max$$

при  $0 \leq x_i \leq 1, i = 1, 2, 3$ .

Простое вычисление показывает, что  $M = (4, 1), N = (1, 3)$ . Тогда  $x(\lambda)$  определяется из решения задачи

$$\frac{3}{x_1 + x_2 + x_3 + 1} \rightarrow \max, \quad \frac{-x_2 + x_3 + 1}{x_1 - 0.75x_2 + 1} = \lambda, \quad 0 \leq x_i \leq 1, i = 1, 2, 3.$$

При

$$\lambda = 2, (F_1(x(2)), F_2(x(2))) = (2, 3/2) = C$$

и при  $\lambda = 3, (F_1(x(3)), F_2(x(3))) = (3, 12/11) = D$ . Линейная свертка не позволяет определить эффективные точки C и D (см.Рис.1).

Однако, отказ от линейной свертки (хотя она не пригодна для полного определения ПГ задачи (1)) часто становится причиной неприменимости эффективных частных методов решения ЛП, если этому позволяет  $X$  (например, задача транспортного типа). Невозможность применения частных методов в случае задачи (5) связана с нелинейностью  $F_2(x)$  и присутствия дополнительного ограничения  $F_1(x) = \lambda$ . Ниже предлагается схема разложения, которая позволяет устранить эту проблему.

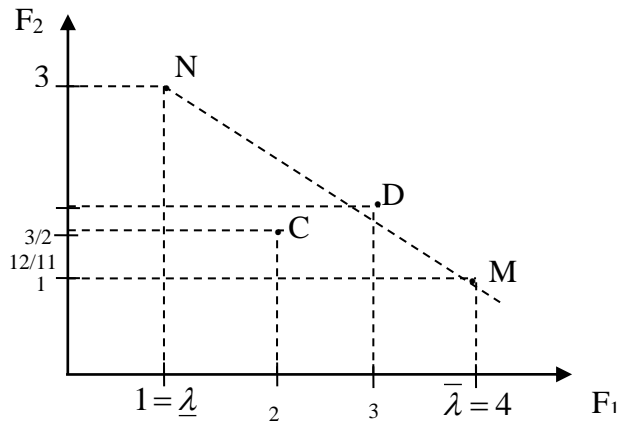


Рис 1.

#### 4. Алгоритм разложения для задачи (5)

Обозначим

$$X(\lambda) = \{x \in X \mid F_1(x) = \lambda\},$$

$$\Phi(\lambda) = \{(f_2(x), g_2(x)) \mid x \in X(\lambda)\}.$$

Множество  $\Phi(\lambda)$  образует многогранник в первой четверти координатной плоскости  $E^2$ .

Алгоритм разложения организуется в двух этапах. На первом этапе рассматривается и решается

**Задача  $P_0$ .** Найти допустимую точку  $x^0 \in X(\lambda)$ .

Решение задачи  $P_0$  организуем в форме следующей процедуры  $\Pi_0$ .

**Описание процедуры  $\Pi_0$ .**

Решаем задачу

$$\varphi = f_1(x) - \lambda g_1(x) \rightarrow \min, x \in X,$$

и задачу

$$\varphi = f_1(x) - \lambda g_1(x) \rightarrow \max, x \in X.$$

Находим их оптимальные решения  $\bar{x}$  и  $\bar{\bar{x}}$ , соответственно. Так как  $\underline{\lambda} < \lambda < \bar{\lambda}$ , имеем  $\varphi(\bar{x}) < 0 < \varphi(\bar{\bar{x}})$ . Определяем число  $\lambda^0$  из решения уравнения  $\varphi(\lambda \bar{x} + (1 - \lambda) \bar{\bar{x}}) = 0$ , существование которого следует из последнего неравенства. Тогда  $x^0 = \lambda \bar{x} + (1 - \lambda) \bar{\bar{x}}$  будет допустимым

искомым решением задачи (5).  $x^0$  позволяет определить точку  $A_0 = (f_2(x^0), g_2(x^0)) \in \Phi$  и вектор  $\vec{n}(x^0) = (\alpha^{(1)}, \beta^{(1)}) = (g_2(x^0), -f_2(x^0))$ . Далее составляем задачу  $\alpha^{(1)} f_2(x) + \beta^{(1)} g_2(x) \rightarrow \max, x \in X(\lambda)$  и завершаем первый этап схемы разложения (см.Рис.2).

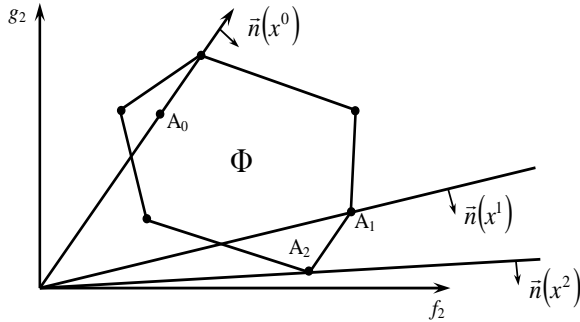


Рис.2

На втором этапе рассматривается и решается

**Задача  $P_i$**

$$\alpha^{(i)} f_2(x) + \beta^{(i)} g_2(x) \rightarrow \max, x \in X(\lambda), \quad (6)$$

при  $i=1,2,\dots$

Задача (6) является задачей ЛП и ее ограничение  $x \in X(\lambda)$  состоит из ограничения  $x \in X$  и одного дополнительного линейного ограничения. Ниже приводим специальную процедуру, которая решает (6) с помощью задач ЛП только с ограничением  $x \in X$  (назовем ее процедурой  $P_1$ ).

При  $i=1$  коэффициенты  $\alpha, \beta$  в (6) вычисляются процедурой  $P_0$  и считаем их известными. Последующие значения этих коэффициентов определяются с помощью оптимального решения  $x^{i-1}$  предыдущей задачи  $P_{i-1}$ :

$$\alpha^{(i)} = g_2(x^{i-1}), \quad \beta^{(i)} = -f_2(x^{i-1}).$$

Составляем вектор  $\vec{n}(x^i) = (\alpha^{(i)}, \beta^{(i)})$  и точку  $A_i = (f_2(x^i), g_2(x^i))$ .

Число решаемых задач (6) зависит от момента  $k$ , когда выполняется условие  $A_{k-1} = A_k$ . Ее назовем условием остановки процесса решения задачи (5). Когда оно имеет место, решение  $x^k$  принимаем за решение задачи (5).

Обоснование подобного заключения следует из свойства точки  $A_i$ : точка  $A_i$  находится на максимальном расстоянии по направлению  $\vec{n}(x^{i-1})$  от

прямой  $OA_{i-1}$ , соединяющей начало координат  $O$  и точку  $A_{i-1} \in \Phi$ . Это свойство является следствием свойства скалярного произведения двух векторов. Оно обеспечивает монотонность итерационного процесса. Процесс конечен, поскольку число экстремальных точек многогранника конечно.

При нарушении условия  $A_{i-1} = A_i$  решаем задачу (6) при новых значениях коэффициентов  $\alpha$  и  $\beta$ :  $\alpha = \alpha^{(i)}$ ,  $\beta = \beta^{(i)}$  и процесс, таким образом, продолжается.

В примере, представленном на рис.2 решаются всего две задачи – задача  $P_1$  и задача  $P_2$ . Задача  $P_2$  определяет оптимальное решение  $x^2$  задачи (5).

**Описание процедуры  $\Pi_2$ .**

Процедура  $\Pi_2$  решает задачу (7), последовательно решая несколько задач ЛП с ограничением  $X$ :

$$\begin{aligned} g &= \alpha f_2(x) + \beta g_2(x) \rightarrow \max, \\ \varphi &= f_1(x) - \lambda g_1(x) = 0, x \in X. \end{aligned} \tag{7}$$

Рассмотрим множество  $G = \{(\varphi(x), g(x)) | x \in X\}$ . Как и в случае  $\Phi$ , множество  $G$  предствляет многогранник в координатной плоскости  $E^2$  (в отличие от  $\Phi$ ,  $G$  не находится в первой четверти этой плоскости). Ось ординат пересекает  $G$ , поскольку  $\varphi(x) = 0, x \in X$ . Тогда из определения  $\underline{\lambda}$  и  $\overline{\lambda}$  непосредственно имеем  $\min_{x \in X} \varphi(x) = \varphi(x^0) < \max_{x \in X} \varphi(x) = \varphi(x^0)$

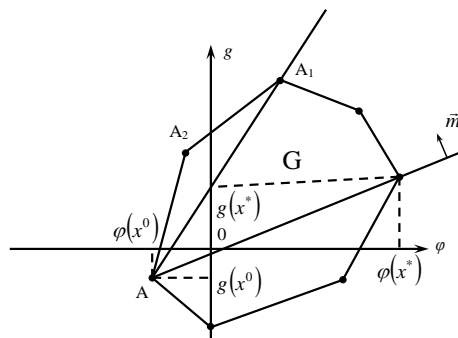


Рис. 3

Шаг 1. Находим точки  $x^0$  и  $x^*$  из решения задач  $\varphi(x) \rightarrow \min, x \in X$  и  $\varphi(x) \rightarrow \max, x \in X$ , соответственно.

Шаг 2. Составляем вектор  $\overrightarrow{AB} = (\alpha, \beta) = (\varphi(x^*) - \varphi(x^0), g(x^*) - g(x^0))$  и вектор  $\vec{m} = (-\beta, \alpha)$ . Очевидно, что  $\overrightarrow{AB} \perp \vec{m}$

Шаг 3. Решаем задачу

$$-\beta \varphi(x) + \alpha g(x) \rightarrow \max, x \in X \quad (8)$$

и находим в  $G$  точку, которая максимально удалена по направлению  $\vec{m}$  от прямой  $AB$  (на рис.3 такую точку представляет  $A_1$ ). Среди всех векторов  $\vec{OK}, K = (\varphi(x), g(x)), x \in X$  вектор  $\vec{OA_1}$  имеет наибольшую проекцию по направлению  $\vec{m}$ . Затем среди пары точек  $(A, A_1)$  и  $(A_1, B)$  выбираем ту пару, у которой абсциссы ее точек имеют разные знаки. В нашем примере это будет пара  $(A, A_1)$  (см.Рис.3).

Шаг4. Вектор  $\vec{AB}$  заменяем вектором  $\vec{AA_1}$ , если они не коллинеарные, то переходим ко второму шагу. В противном случае выполняем следующий шаг 5.

В примере, представленном на рис.3, выполнение указанных шагов сначала переводит вектор  $\vec{AB}$  на вектор  $\vec{AA_1}$ , а затем на вектор  $\vec{A_2A_1}$ . Определением  $\vec{A_2A_1}$  выполнение этих шагов завершается и решение задачи (7) определяем следующим образом: сначала находим число  $\lambda^0$  из уравнения  $(\lambda A_2 + (1 - \lambda)A_1)_\varphi = 0$  и берем решения  $x_{A_2}$  и  $x_{A_1}$  задачи (8), которые определяют точки  $A_2 = (\varphi(x_{A_2}), g(x_{A_2}))$  и  $A_1 = (\varphi(x_{A_1}), g(x_{A_1}))$ , соответственно.

Далее, решение задачи (7) берем в виде выпуклой комбинации  $\lambda_0 x_{A_2} + (1 - \lambda)x_{A_1}$ .

Шаг5. Берем решения  $x_B$  и  $x_A$ , которые определяют точки  $B$  и  $A$ , соответственно, находим число  $\lambda^0$  из уравнения  $(\lambda B + (1 - \lambda)A)_\varphi = 0$  и, наконец, решение (7) находим в виде выпуклой комбинации  $\lambda^0 x_B + (1 - \lambda^0)x_A$ .

## 5. Диалоговый алгоритм принятия решения

Не всегда есть необходимость полностью построить ПГ с целью принятия окончательного решения. Достаточно направить процесс аппроксимации в ту часть ПГ, где лицо, принимающее решение, считает целесообразным продолжить поиск окончательного решения.

### *Описание алгоритма*

Шаг 1. Сначала представляем ЛПР крайние точки ПГ и предлагаем произвести выбор среди них. Если выбор завершается принятием одного из этих вариантов, как «хорошее решение», делаем переход на шаг3. В противном случае, ЛПР должно отойти от этих точек вдоль паретовой границы  $Y^P$ . Так как ПГ представляет график монотонной непрерывной



функции от аргумента  $\lambda \in [\underline{\lambda}, \bar{\lambda}]$ , то мы можем организовать это отклонение с помощью следующего шага.

Шаг 2. Находим новую точку  $y^1 \in Y^P$ , для которой  $y_1^1 = \frac{\lambda + \bar{\lambda}}{2}$  и

представляем  $y^1$  ЛПР. После этого ЛПР исключает из дальнейшего рассмотрения или точку  $A$  или точку  $B$ . Он продолжает поиск на участке ПГ при  $\lambda \in [\underline{\lambda}, y_1^1]$ , если исключает  $B$  (тогда полагаем  $B = y^1$ ), или на участке ПГ при  $\lambda \in [y_1^1, \bar{\lambda}]$ , если исключает  $A$  (тогда полагаем  $A = y^1$ ). Таким образом, область поиска окончательного решения сужается. Теперь делаем переход на первый шаг и его повторно выполняем.

Шаг 3. Выбранный вариант из  $A$  и  $B$  принимаем как окончательное решение и завершаем процесс поиска.

### Литература

1. Hamidov R.H., An algorithm for equilibrium point of one N-person non cooperative game, The 38<sup>th</sup> Annual Iranian Mathematical Conference, Tehran, 2007, p.10-14.
2. Polak E., On the approximation of solutions to multiple criteria decision, Making, Kyoto, 1975, Springer-Verlag, 1976, pp.271-282.
3. Гамидов Р.Г., Кусочно-линейная аппроксимация множества эффективных оценок двукритериальных задач, М.: ВИНТИ, 1983, №6414, 11с.
4. Гамидов Р.Г., Метод разложения для задачи математического программирования, Известия НАН Азербайджана, серия физ.-техн. и мат. наук, том XV111, №5, 2005, с.32-34.
5. Полищук Л.И., Кусочно-линейная аппроксимация паретовой границы выпуклых двукритериальных задач, В кн. Модели и методы исследования экономических систем, Новосибирск, Наука, 1979, с.108-116.

## **Kəsr-xətti proqramlaşdırmanın ikikriteriyalı məsələsində qərar qəbuletmə**

**Həmidov R.H., Allahverdiyeva N.K.**

### **XÜLASƏ**

Təqdim olunan işdə ikikriteriyalı kəsr-xətti proqramlaşdırma məsələsinə baxılır. Pareto sərhəddinin qurulması üsulu və Pareto sərhəddində son kompromis həllin axtarılması alqoritmi təklif olunur. Axtarış qərar qəbul edən şəxsdən alınan əlavə informasiya əsasında həyata keçirilir. Daha sonra qərar qəbuletmə prosesinin böyük ölçülü məsələlərə tətbiqinin mümkün olması üçün ayrılış sxemi təklif olunur. Təklif olunan üsulun yerinə yetirilməsi gedisinin qrafik təsviri verilir.

**Açar sözlər:** xətti proqramlaşdırma, kəsr-xətti funksiya, effektiv həll, simpleks üsul, böyük ölçülü məsələ, ayrılış alqoritmi, Pareto sərhəddi.

## **Decision-making in problem of the bicriterial fractional linear programming**

**Hamidov R.H., Allahverdiyeva N.K.**

### **ABSTRACT**

Bicriterion fractional linear programming is considered. A procedure to identify all Pareto optimal points is analyzed. We use decision maker's additional information to make final choice in the set of all Pareto points. Then we present new decomposition scheme to make the suggested decision making process available for the large scale problems. We use graphical representation of course of separate stage of the process.

**Keywords:** linear programming, bilinear function, effective solution, simplex method, large-scale problem, decomposition algorithm, Pareto bound.