электронное научно-техническое издание
НАУКА и ОБРАЗОВАНИЕ
Эл № ФС 77 - 30569. Государственная регистрация №0421100025. ISSN 1994-0408

Популяционные методы аппроксимации множества Парето в задаче многокритериальной оптимизации. Обзор.

77-30569/363023

04, апрель 2012

Карпенко А. П., Семенихин А. С., Митина Е. В.

УДК.519.6

МГТУ им. Н.Э. Баумана
apkarpenko@mail.ru
saspost@yandex.ru
mitinakaterina@gmail.com

Введение

Известно большое число методов решения задачи многокритериальной оптимизации (МКО-задачи), из числа которых перспективными являются методы, основанные на предварительном построении аппроксимации множества Парето (а тем самым, и фронта Парето) этой задачи [1, 2]. Простейшим из таких методов является сеточный метод [3]. В ситуации, когда требуется высокая точность аппроксимации множеств Парето и/или когда имеет место высокая вычислительная сложность целевых функций, сеточный метод может требовать неприемлемо высоких вычислительных ресурсов. Поэтому в настоящее время интенсивно развиваются альтернативные методы [4].

Обычно указанные методы строят на основе эволюционных алгоритмов и чаще всего – на основе генетических алгоритмов [5]. При этом соответствующие методы Парето-аппроксимации называют эволюционными. Обзор таких методов представлен, например, в работах [6, 7].

Принципиальным в этих методах является не использование именно эволюционных алгоритмов, а правила формирования фитнесс-функции, обеспечивающие перемещение индивидов популяции, в конечном счете, в направлении множества Парето. Эволюция же этих индивидов может протекать по законам, отличным от законов, используемых в эволюционных алгоритмах, например, по законам миграции частиц в алгоритме роя частиц [8, 9]. Поэтому в качестве общего названия рассматриваемых методов используем термин «популяционные методы Парето-аппроксимации».

Для полноты картины, наряду с популяционными методами в работе рассматриваем наиболее известные не популяционные методы. Особенностью обзора является то, что в той его части, которая посвящена популяционным методам, мы концентрируем наше внимание, преимущественно, на правилах формирования фитнесс-функции, используемых в этих методах.

Можно выделить следующие основные требования к методам Парето-аппроксимации [7]:

- точки найденной аппроксимации расположены достаточно близко к точному множеству Парето;

- аппроксимация покрывает все множество Парето;

- распределение точек аппроксимации равномерно.

В современных методах Парето-аппроксимации для выполнения последнего требования используют специальные механизмы, обеспечивающие приемлемый разброс (*spread*) точек аппроксимации. Наиболее известным механизмом такого сорта является механизм нишевания (*niching*) [10]. Самостоятельную проблему представляет разработка критериев оценки качества Парето-аппроксимации, которые отражали бы указанные требования. Примеры таких критериев рассмотрены, например, в работах [11, 12].

Известно несколько классификаций методов Парето-аппроксимации. Используем классификацию, предложенную Гуменниковой А. В. [13], дополнив ее классом «наивных» методов [14] и классом прочих методов. Итого рассматриваем следующие пять классов методов Парето-аппроксимации:

- «наивные» методы;
- методы переключающихся целевых функций;
- методы агрегации целевых функций;
- методы на основе ранжирования агентов популяции;
- прочие методы.

В разделе 1 приведена математическая постановка задачи, разделы 2 - 6 посвящены рассмотрению указанных классов методов Парето-аппроксимации. В заключении сформулированы основные выводы.

1. Постановка задачи и общая схема популяционных алгоритмов ее решения

Совокупность частных критериев оптимальности (частных целевых функций) $f_k(X),\ k \in [1:|F|]$ образует векторный критерий оптимальности (векторную целевую функцию) $F(X) \in \{F\}$, где $X \in \{X\}$ - вектор варьируемых параметров; $\{X\}$, $\{F\}$ - пространства параметров и критериев соответственно. Здесь и далее запись вида $|A|$, где A – некоторый вектор или счетное множество, означает размерность этих объектов. Положим, что ставится задача минимизации каждого из частных критериев в одной и той же области допустимых значений $D_X \subset R^n$. Тогда во введенных обозначениях МКО-задачу *условно* можно записать в виде

$$\min_{X \in D_X} F(X) = F(X^*) = F^*, \tag{1}$$

где X^*, F^* - решения задачи.

Полагаем, что частные критерии оптимальности нормализованы, например, по формуле

$$f_k(X) = \frac{f_k(X) - f_k^*}{f_k^{**} - f_k^*}, k \in [1:|F|],$$

где $f_k^* = \min f_k(X)$, $f_k^{**} = \max f_k(X)$, $X \in D_X$.

Векторный критерий оптимальности $F(X)$ выполняет отображение области D_X в некоторое множество D_F, называемое множеством достижимости МКО-задачи (1). Введем на множестве D_X отношение предпочтения $\succ$. Будем говорить, что вектор $X_1 \in D_X$ предпочтительнее вектора $X_2 \in D_X$ и писать $X_1 \succ X_2$, если среди равенств и неравенств $f_k(X_1) \le f_k(X_2)$, $k \in [1:|F|]$ имеется, хотя бы одно строгое неравенство. Аналогично, на множестве D_F введем отношение доминирования: будем говорить, что вектор $F(X_1) \in D_F$ доминирует вектор $F(X_2) \in D_F$, и писать $F(X_1) \rhd F(X_2)$, если $X_1 \succ X_2$.

Выделим из множества D_F подмножество D_F^* точек, среди которых нет доминируемых. Это множество называют *фронтом Парето*. Множество $D_X^* \in D_X$, соответствующее множеству D_F^*, называют *множеством Парето* (переговорным множеством, областью компромисса). Определения множества и фронта Парето иллюстрирует рисунок 1. Поскольку множество D_F на этом рисунке является выпуклым, фронт Парето D_F^* в данном случае представляет собой дугу AB, на которой точка A соответствует f_1^*, а точка B — f_2^*. Среди точек $F(X_1)$, $F(X_2)$, лежащих на фронте Парето, нет доминируемых, поскольку если $f_1(X_1) < f_1(X_2)$, то обязательно $f_2(X_1) > f_2(X_2)$.

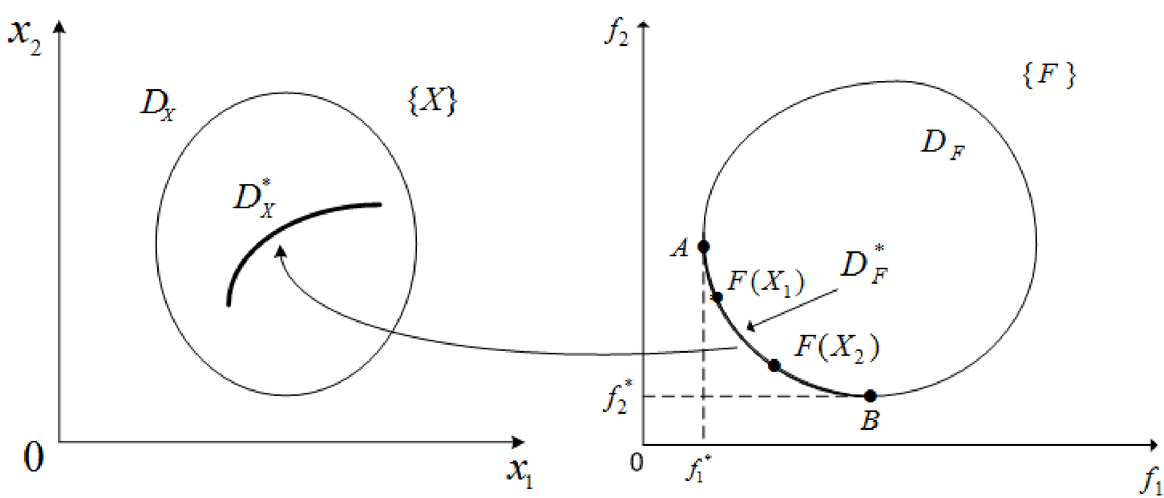

Рисунок 1 - К определению множества Парето: $|X| = 2$; $|F| = 2$

Ставим задачу приближенного построения множества Парето (а, тем самым, и фронта Парето) в МКО-задаче (1). Называем эту задачу *задачей Парето-аппроксимации*.

Пусть тем или иным образом уже сформировано *архивное множество* A_F, содержащее не доминируемые точки F_j^A, а также архивное множество A_X соответствующих ему точек $X_j^A;\ j \in [1:|A|]$. Здесь $|A|$ - мощность множеств A_F, A_X.

Суть большинства популяционных методов Парето-аппроксимации состоит в итерационном уточнении множеств точек в архивах A_F, A_X. Если при этом на итерации t появляется новая точка F_i, доминирующая некоторые точки из архива A_F, то все доминируемые точки, а также соответствующие точки из архива A_X, удаляем. При удовлетворении некоторого критерия останова текущее содержимое архивов A_F, A_X полагаем искомой аппроксимацией фронта Парето D_F^* и множества Парето D_X^* соответственно.

В популяционных методах Парето-аппроксимации новые точки для архивов A_F, A_X «поставляет» популяция агентов s_i, текущие координаты которых в пространстве поиска $\{X\}$ равны X_i, а в пространстве $\{F\}$ - $F_i = F(X_i);\ i \in [1:|S|]$.

В популяционных оптимизационных алгоритмах миграция агентов в пространстве поиска подчинена задаче минимизации (для определенности) значений фитнесс-функции. Основной проблемой построения популяционных методов Парето-аппроксимации является построение фитнесс-функции $\varphi(X)$, обеспечивающей перемещение агентов популяции s_i, $i \in [1:|S|]$ в направлении множества Парето D_X^*, а соответствующих точек F_i - в направлении фронта Парето D_F^*.

В силу, как правило, меньшей размерности критериального пространства $\{F\}$ по сравнению с размерностью пространства поиска $\{X\}$, ответ на вопрос о направлении и шаге перемещения агентов обычно отыскивают в терминах критериального пространства, а не пространства параметров. Важно также, что относительно множества Парето, по сути, нет никой априорной информации, кроме того, что это множество точек, не связанных между собой отношением предпочтения $\succ$. В то же время по отношению к фронту Парето априорной информации значительно больше. Например, с учетом того, что частные критерии оптимальности нормированы и поэтому $f_k(X) \in [0;1],\ k \in [1:|F|]$, имеет место

Утверждение 1. Любой луч, проведенный из начала системы координат $0 f_1\ f_2 \ldots f_{|F|}$ в неотрицательном направлении осей $0f_1,\ 0f_2, \ldots, 0f_{|F|}$ пересекает фронт Парето не более чем в одной точке.

Доказательство проведем от противного. Допустим, что утверждение неверно, и указанный луч пересекает фронт Парето в точках F_1^*, F_2^*. В этом случае, по крайней мере, для одного $k \in [1:|F|]$ должно быть справедливо неравенство $f_{1,k}^* < f_{2,k}^*$ или неравенство $f_{1,k}^* > f_{2,k}^*$, что противоречит условию принадлежности точек F_1^*, F_2^* фронту Парето●

Выделяют четыре класса задач Парето-оптимизации, соответствующих следующим четырем типам фронта Парето: выпуклые задачи; вогнутые задачи; выпукло-вогнутые задачи; разрывные задачи (рисунок 2).

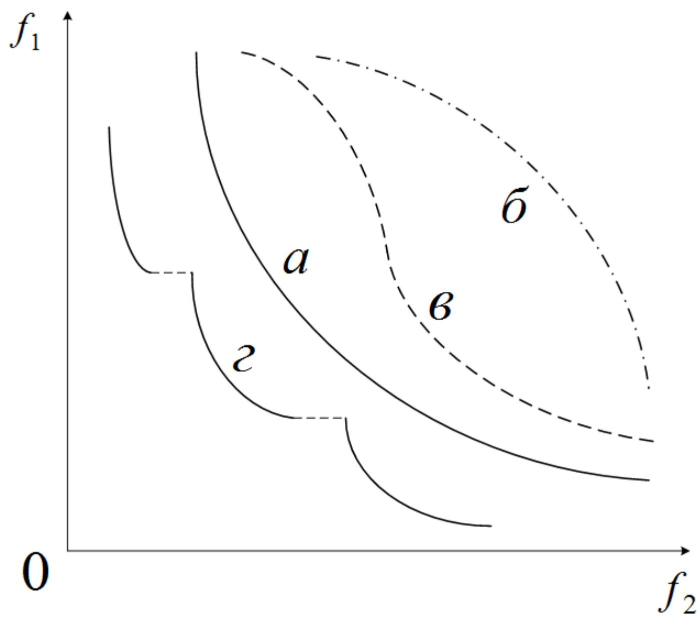

Рисунок 2 – Типы фронтов Парето ($|F| = 2$): а) выпуклый; б) вогнутый; в) выпукло-вогнутый фронт; г) разрывный

Одной из проблем популяционных методов Парето-аппроксимации является формирование начальных состояний архивов A_F, A_X. С этой целью может быть использован сеточный метод, суть которого состоит в следующем. Покрываем область D_X некоторой сеткой с узлами X_i, $i \in [1:n]$. В каждом из этих узлов вычисляем значение вектор-функции F_i, среди векторов F_i выбираем недоминируемые векторы $\{F_{i_j}^A\}$ и находим соответствующее им множество точек $\{X_{i_j}^A\}$, где $F_{i_j}^A = F(X_{i_j}^A)$, $i_j \in [1:|A|]$. Множества $\{X_{i_j}^A\}$, $\{F_{i_j}^A\}$ представляют собой искомую начальную дискретную аппроксимацию множества Парето и фронта Парето МКО-задачи (1) соответственно.

Известным вариантом сеточного метода является метод исследования пространства параметров [15], особенностью которого является использование специальных сеток, построенных на основе так называемых $ЛП_\tau$ последовательностей, обеспечивающих большую репрезентативность Парето-аппроксимации.

2. «Наивные» методы

Данный класс методов выделяет Люк (*S. Luke*) в своей известной книге [14]. Простейшим методом данного класса является *метод на основе лексикографической турнирной селекции* (*lexicographic tournament selection*).

Метод исходит из того, что частные критерии $f_k(X),\ k \in [1:|F|]$ упорядочены по важности, так что самым важным является критерий $f_1(X)$, следующим по важности – критерий $f_2(X)$ и так далее. Рассмотрим правило сравнения приспособленностей агентов, используемое данным методом. Пусть $s_i, s_j,\ i, j \in [1:|S|], i \neq j$ - два сравниваемых агента в текущих состояниях X_i, X_j. Лучшего из этих агентов определяем путем последовательного сравнения пар величин $\left(f_k(X_i),\ f_k(X_j)\right),\ k = 1, 2, ..., |F|$ до тех пор, пока не будет установлено, что, например, $f_{k*}(X_i) < f_{k*}(X_j)$. В таком случае полагаем, что агент s_i имеет лучшую приспособленность по сравнению с агентом s_j.

Лучшего агента $s^{best} \in S$ популяции в ее текущем состоянии $X_i,\ i \in [1:|S|]$ определяем в результате следующей последовательности шагов.

1) Выбираем случайного агента популяции s_{i_1} и полагаем его лучшим агентом, т.е. полагаем $X^{best} = X_{i_1}$. Выполняем присваивание $j = 2$.

2) Выбираем другого случайного агента s_{i_j}.

3) По указанному выше правилу определяем лучшего из агентов s^{best}, s_{i_j} и объявляем его новым агентом s^{best}.

4) Если $j < m$ (размер турнира m не исчерпан), то полагаем $j = j + 1$ и возвращаемся к шагу 2.

Известен ряд модификаций рассмотренного метода. Так для выбора лучшего агента можно использовать случайный критерий из числа критериев $f_k, k \in [1:|F|]$. Можно использовать процедуру голосования, когда лучшим считается агент, которому соответствуют наименьшие значения наибольшего числа критериев. Наконец, можно использовать многоуровневую турнирную селекцию (в случае $|F| > 2$), когда основной турнир проводим на основании одного критерия, однако агентов для этого турнира отбираем с использованием турнира по другому критерию, агентов для этого турнира отбираем с использованием турнира по третьему критерию и т.д.

3. Методы переключающихся целевых функций

В данных методах выбор лучших агентов популяции производится на основе сравнения соответствующих значений различных частных критериев оптимальности. Наиболее известным алгоритмом Парето-аппроксимации, основанном на *методе* *переключающихся целевых функций*, является алгоритм *VEGA* (*Vector Evaluated Genetic Algorithm*), предложенный Шафером (*D. Shaffer*) в 1985 г. [16].

Пусть $X_i,\ i \in [1:|S|]$ текущие положения агентов популяции, где $|S| > |F|$ и эти величины кратны. Пригодность агентов в алгоритме *VEGA* определяют по следующей схеме.

1) В соответствии с принятым правилом селекции, основываясь на фитнесс-функции $\varphi_1(X) = f_1(X)$, выбираем $\frac{|S|}{|F|}$ лучших агентов, не исключая их из популяции.

2) Аналогично, основываясь на фитнесс-функции $\varphi_2(X) = f_2(X)$, выбираем следующие $\frac{|S|}{|F|}$ лучших агентов.

….

$|F|$) Основываясь на фитнесс-функции $\varphi_{|F|}(X) = f_{|F|}(X)$, выбираем $\frac{|S|}{|F|}$ последних лучших агентов.

В качестве правила селекции алгоритм *VEGA* использует правило рулетки, когда вероятность выбора агента s_i пропорциональна его относительной приспособленности (если речь идет о максимизации целевых функций):

$$\frac{\varphi(X_i)}{\sum_j \varphi(X_j),\ \ j \in [1:|S|]},\quad i \in [1:|S|].$$

Рассмотренную схему отбора иллюстрирует рисунок 3, на котором представлены проекции точек $F_i,\ i \in [1:|S|]$ на плоскость $0 f_1 f_2$ пространства критериев $\{F\}$. Показаны множества агентов $S_1,\ S_2$, отобранных на основе критериев оптимальности $f_1(X),\ f_2(X)$ соответственно (имеются в виду задача многокритериальной максимизации).

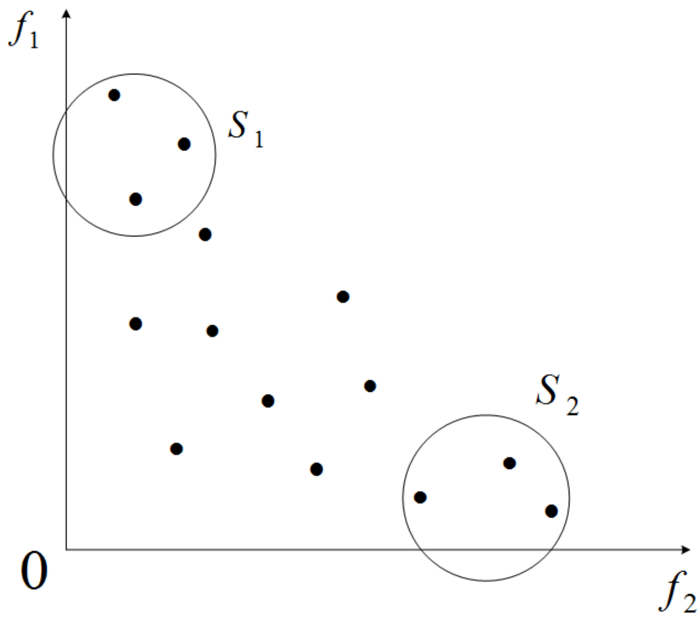

Рисунок 3 – К схеме алгоритма *VEGA*

4. Методы агрегации целевых функций

В данном случае целевые функции агрегируют (сворачивают) в один параметризованный скалярный критерий, который выступает в роли фитнесс-функции. Рассматриваем следующие методы данного класса: метод взвешенных критериев; метод идеальной точки; адаптивный метод взвешенных сумм.

4.1. Метод взвешенных критериев

Теоретической основой *метода взвешенных критериев* (*Sum of Weighted Objectives, SWO*) является следующая известная теорема [1, 2].

Теорема 1. Если для некоторых весовых множителей $\lambda_k > 0$, $k \in [1:|F|]$ имеет место равенство

$$\min_{X \in D_X} \sum_{k=1}^{|F|} \lambda_k \; f_k(X) = \sum_{k=1}^{|F|} \lambda_k \; f_k(X^*), \tag{2}$$

где $X^* \in D_X$, то вектор X^* оптимален по Парето.

Доказательство. Пусть вектор X^* не оптимален по Парето. Тогда существует такой вектор $X \in D_X$, что

$$f_k X) \le f_k(X^*), \quad k \in [1:|F|],$$

причем хотя бы одно из неравенств является строгим. Умножая каждое из последних неравенств на $\lambda_k > 0$ и складывая, получим

$$\sum_{k=1}^{|F|} \lambda_k \ f_k(X) < \sum_{k=1}^{|F|} \lambda_k \ f_k(X^*),$$

что противоречит условию теоремы●

Теорема 1 показывает, что выбор определенной точки из множества Парето эквивалентен указанию весов для каждого из частных критериев оптимальности. Важно, что теорема задает лишь *необходимое условие* оптимальности по Парето вектора $X^* \in D_X$. Т.е. из того факта, что точка X^* принадлежит множеству Парето, не следует, что эта точка обязательно удовлетворяет условию (2) – случай невыпуклого множества D_F^* (рисунок 4).

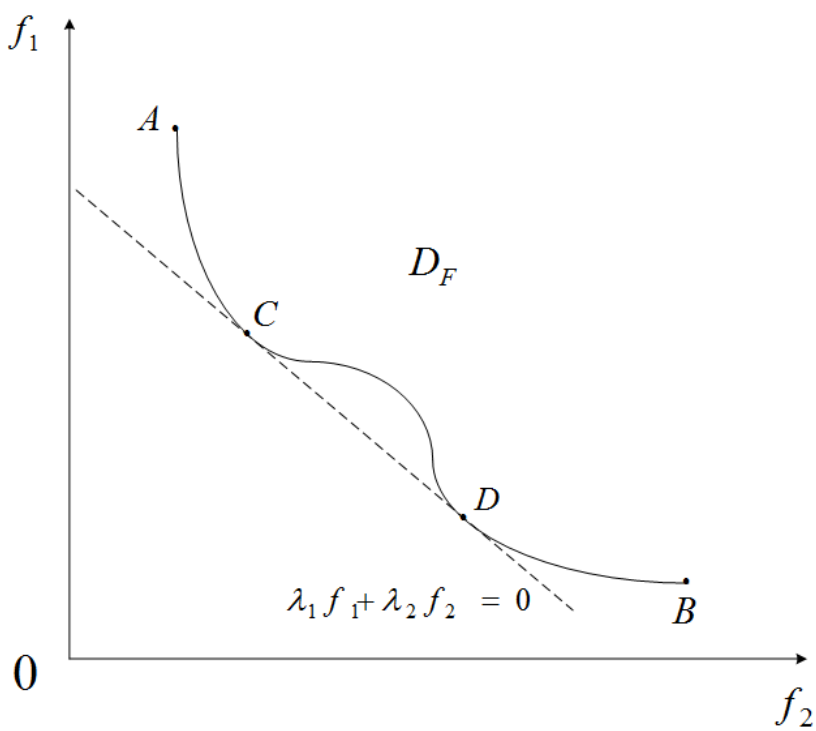

Рисунок 4 – Невыпуклый фронт Парето: $|F| = 2$; дуга AB - фронт Парето D_F^*

В качестве фитнесс-функции в методе взвешенных критериев используем функцию

$$\varphi(X) = \varphi(\Lambda, F) = \sum_{k=1}^{|F|} \lambda_k \ f_k(X),$$

где $(|F| \times 1)$-вектор $\Lambda = (\lambda_1, \lambda_2, ..., \lambda_{|F|})$. Основным недостатком метода является невозможность с его помощью локализовать точки фронта Парето, принадлежащие его невыпуклой части (дуга CD на рисунке 4).

Метод широко используется для построения не популяционных алгоритмов Парето-аппроксимации, схема которых состоит из двух следующих основных шагов.

1) Множество $D_\Lambda = (\Lambda \mid 0 \le \lambda_i \le 1,\ i \in [1:|F|])$ допустимых значений вектора весовых множителей $\Lambda = (\lambda_1, \lambda_2, ..., \lambda_{|F|})$ покрываем некоторой сеткой с узлами $\Lambda_j = (\lambda_{1,j}, \lambda_{2,j}, ..., \lambda_{|F|,j})$, $j = 1,2,...$

2) При каждом Λ_j решаем задачу глобальной условной оптимизации (2) – получают точку $X_j^* \in D_X^*$.

4.2. Метод идеальной точки

Идеальной называют точку в пространстве критериев $F^{**} = (f_1^*,\ f_2^*, ..., f_{|F|}^*)$, где

$$f_k^* = f_k(X_k^*) = \min_{X \in D_X} f_k(X), \quad k \in [1:|F|].$$

В качестве фитнесс-функции в этом случае используют функцию

$$\varphi(X) = \rho(F^{**}, F), \tag{3}$$

где $\rho(\cdot, \cdot)$ - некоторая метрика пространства $\{F\}$, например, чебышевская метрика [17]

$$\rho(F^{**}, F) = \max_{k \in [1:|F|]} \left(\lambda_k\ abs(f_k^* - f_k)\right), \ \lambda_k > 0.$$

Здесь $abs(\cdot)$ - символ абсолютного значения числа.

Метод идеальной точки иллюстрирует рисунок 5, показывающий, что, в отличие от метода взвешенных критериев, данный метод позволяет отыскивать точки, лежащие на невыпуклой части фронта Парето. Заметим, что свертка на основе идеальной точки (3) близка к свертке Гермейера [17].

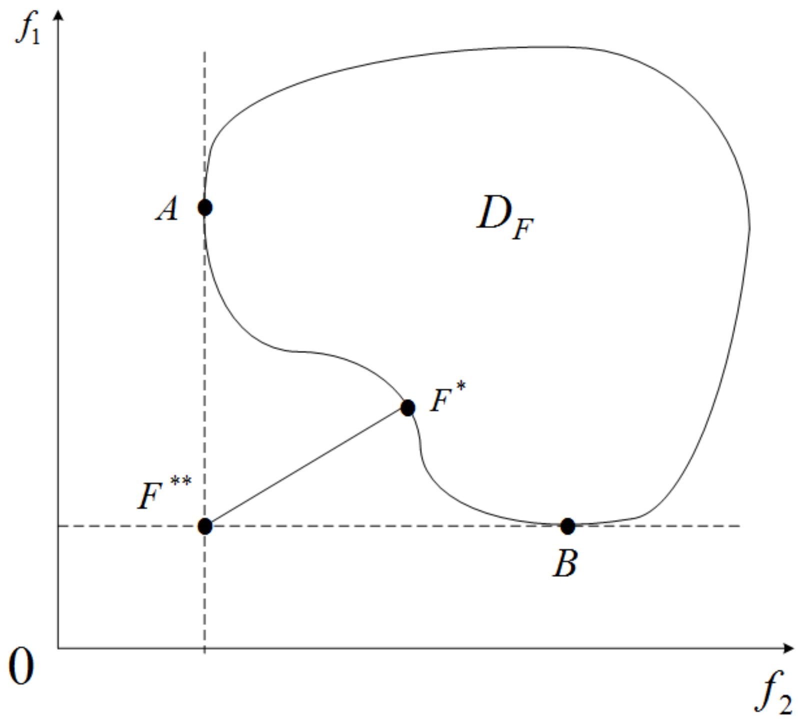

Рисунок 5 – К методу идеальной точки: $|F| = 2$; дуга AB - фронт Парето

Метод может быть использован для построения не популяционных алгоритмов Парето-аппроксимации. Задачу Парето-аппроксимации в этом случае сводят к многократному решению задачи глобальной оптимизации

$$\min_{X \in D_X} \rho(F^{**}, F) = \rho(F^{**}, F^*)$$

при различных допустимых значениях компонентов вектора весов $\Lambda = (\lambda_1, \lambda_2, ..., \lambda_{|F|})$ по схеме, близкой к схеме, приведенной в предыдущем пункте.

4.3. Адаптивный метод взвешенных сумм

Адаптивный метод взвешенных сумм (*Adaptive Weighted Sum Method, AWSM*) предложили Рю, Ким и Ван (*J-H. Ryu, S. Kim, H. Wan*) в 2009 г. [18]. Целью разработки метода было преодоление указанного выше ограничения метода взвешенных критериев, заключающегося в невозможности отыскания точек, принадлежащих невыпуклым частям фронта Парето. Метод не является популяционным. Мы включаем его в обзор вследствие новизны и высокого потенциала развития.

Метод разработан для двухкритериальной задачи ($|F| = 2$) и включает в себя три следующие основные процедуры:

- определение центральной точки;
- формирование метамоделей частных критериев;
- решение полученных оптимизационных задач.

Рассмотрим суть указанных процедур.

Определение центральной точки. На этапе инициализации центральная точка X_C^0 выбирается случайным образом в области D_X. На этом же этапе должны быть определены следующие свободные параметры метода: $\delta^0 = \delta$ - начальный *радиус области доверия* (*trust region radius*); $\rho \in (0;1)$ - коэффициент сужения области доверия; $\delta_{\min}$ - минимальная величина радиуса области доверия.

На итерации $t+1$ центральная точка X_C' отыскивается среди точек текущей Парето-аппроксимации $A_X = A_X^t$, построенной на предыдущей итерации t.

Отсортируем элементы архивных множеств A_X, A_F по возрастанию первого частного критерия оптимальности $f_1(X)$ и представим в виде линейных списков с прежними наименованиями. Определим расстояние d_j архивной точки F_j^A до ближайших к ней в списке A_F точек формулой

$$d_j = \| F_{j-1}^A - F_j^A \| + \| F_j^A - F_{j+1}^A |, \quad 1 < j < |A|, \tag{4}$$

где $\| \cdot \|$ - символ евклидовой нормы.

Метод использует следующее правило определения центральной точки X_C'.

1) Если $|A| > 2$, то полагаем $X_C' = X_{j^*}^A$, где

$$j^* = \arg\left(\max_{j \in [2:(|A|-1)]} d_j \mid X_j^A \notin X_T \right).$$

Здесь X_T - множество точек, использованных в качестве центральных на всех предыдущих итерациях $[0:\ t]$. Иными словами, за центральную точку принимаем точку, во-первых, наиболее удаленную от других точек множества A_X в смысле расстояния (4), и, во-вторых, не использованную на предшествующих итерациях.

2) Если $|A| = 2$, то с равной вероятностью полагаем $X_C' = X_1^A$ или $X_C' = X_2^A$.

3) Если $|A| = 1$, то принимаем $X_C' = X_1^A$.

Формирование метамоделей. Метамодели представляют собой квадратичные аппроксимации $m_1'(X)$, $m_2'(X)$ функций $f_1(X)$, $f_2(X)$ в окрестности точки X_C':

$$m_1'(X) = f_1(X_C') + G_1^T T(X_C')(X - X_C') + \frac{1}{2}(X - X_C')^T H_1(X_C')(X - X_C');$$

$$m_2'(X) = f_2(X_C') + G_2^T(X_C')(X - X_C') + \frac{1}{2}(X - X_C')^T H_2(X_C')(X - X_C').$$

Здесь ^{T} - символ транспонирования; $G_1(X_C')$, $G_2(X_C')$ - градиенты функций $f_1(X)$,

$f_2(X)$ в точке X'_C соответственно; $H_1(X'_C)$, $H_2(X'_C)$ - матрицы Гессе этих функций.

Если $|A| > 2$, то дополнительно строим метамодели

$$m'_p(X) = \lambda_1^p \, m'_1(X) + \lambda_2^p \, m'_2(X),$$

$$m'_q(X) = \lambda_1^q \, m'_1(X) + \lambda_2^q \, m'_2(X),$$

а если $|A| = 2$ или $|A| = 1$ - метамодель

$$m'_p(X) = \lambda_1^p \, m'_1(X) + \lambda_2^p \, m'_2(X).$$

В первом случае ($|A| > 2$) весовые множители $(\lambda_1^p, \lambda_2^p) = \Lambda^p$, $(\lambda_1^q, \lambda_2^q) = \Lambda^q$ определяем по правилу

$$\Lambda^p = c^p \Big((-f_2(X_C^A) + f_2(X_{j^*-1}^A)),\ (f_1(X_C^A) - f_1(X_{j^*-1}^A))\Big),$$

$$\Lambda^q = c^q \Big((-f_2(X_{j^*+1}^A) + f_2(X_C^A)),\ (f_1(X_{j^*+1}^A) - f_1(X_C^A))\Big);$$

во втором случае ($|A| = 2$) – по правилу

$$\Lambda^p = c^p \Big((-f_2(X_2^A) + f_2(X_1^A)),\ (f_1(X_2^A) - f_1(X_1^A))\Big);$$

в третьем случае (когда $|A| = 1$) – по правилу $\Lambda^p = (0{,}5,\ 0{,}5)$. Константы c^p, c^q выбираем таким образом, чтобы обеспечить выполнение условий нормировки $\lambda_1^p + \lambda_2^p = \lambda_1^q + \lambda_2^q = 1$.

Отметим, что при построении метамоделей $m'_1(X)$, $m'_2(X)$ речь может идти не о градиентах и матрице Гессе функций $f_1(X)$, $f_2(X)$, а об их оценках, полученных, например, численными методами (путем соответствующих конечно-разностных аппроксимаций указанных функций).

Решение оптимизационных задач. Данная процедура предполагает решение задач оптимизации

$$\min_{X \in D'_C} m'_1(X) = m'_1(\hat{X}'_1), \quad \min_{X \in D'_C} m'_2(X) = m'_2(\hat{X}'_2), \tag{5}$$

где текущая область доверия D'_C определяет формула

$$D'_C = \{X \mid X \in D_X, \left\| X - X'_C \right\| \le \delta\}.$$

Если $|A| > 2$, то решения $\hat{X}'_1$, $\hat{X}'_2$ задач (5) позволяют отыскать приближенно оптимальные по Парето точки $\hat{X}'_p$, $\hat{X}'_q$, принадлежащие области доверия D'_C, путем решения оптимизационных задач

$$\min_{X \in D'_C} m'_p(X) = m'_p(\hat{X}'_p), \quad \min_{X \in D'_C} m'_q(X) = m'_q(\hat{X}'_q). \qquad (6)$$

Данный этап метода *AWSM* иллюстрирует рисунок 6, на котором принято что $F'_C = F(X'_C)$, $F^A_{j^*-1} = F(X^A_{j^*-1})$, $F^A_{j^*+1} = F(X^A_{j^*+1})$. Отметим, что задачи (5), (6) представляют собой задачи оптимизации квадратичных функций, для решения которых известны высокоэффективные методы, алгоритмы и соответствующее программное обеспечение.

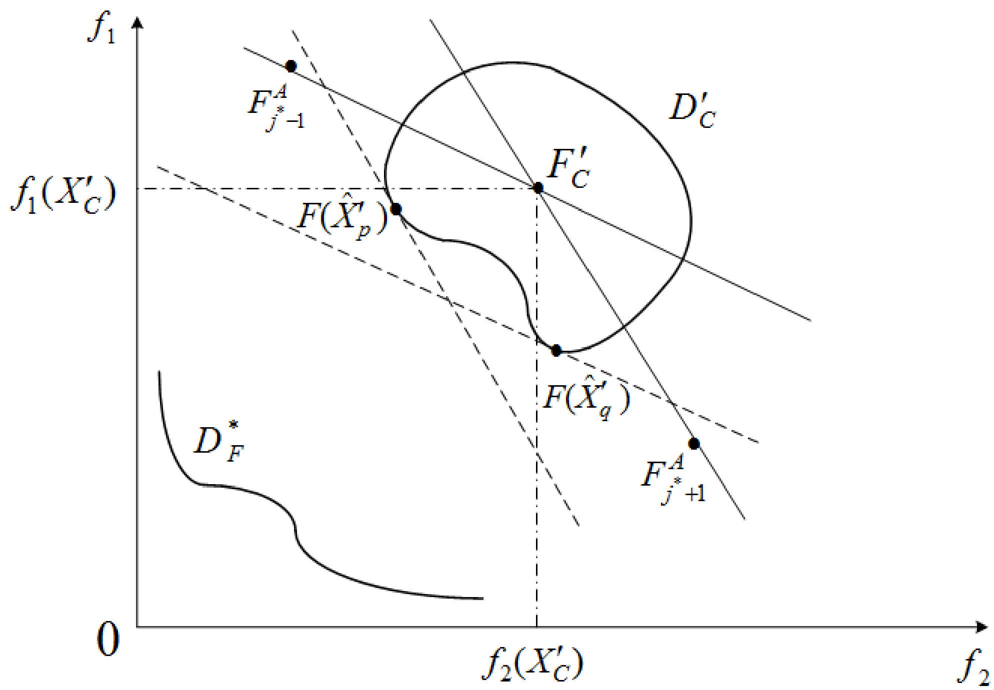

Рисунок 6 – К схеме адаптивного метода взвешенных сумм: результаты решения задач (6)

В процессе итераций текущий радиус области доверия уменьшают по правилу $\delta = \rho\,\delta$ до достижения минимально допустимой его величины $\delta_{\min}$. Новое состояние архивного множества A'_X получаем путем добавления в текущее множество A_X точек $\hat{X}'_1$, $\hat{X}'_2$, $\hat{X}'_p$, $\hat{X}'_q$ и исключения из полученного набора доминируемых решений. Множество D'_F образуют точки, соответствующие построенному множеству A'_X.

5. Методы на основе ранжирования агентов популяции

Основным понятием методов данного класса является понятие *ранга агента популяции*. Известно несколько правил вычисления рангов. Ниже рассмотрены основные из этих правил, используемые в методе недоминируемой сортировки, методе Парето-силы, а также в некоторых других методах.

5.1. Недоминируемая сортировка

Метод недоминируемой сортировки (*Non-Dominated Sorting*, *NDS*) впервые был опубликован в работе Шринивaса (*N. Srinivas*) и Деба (*K. Deb*) в 1994 г. [19]. Метод лежит в основе широко известного генетического алгоритма Парето-аппроксимации *NGSA* (*Non-Dominated Sorting Genetic Algorithm*) [21].

Положим, что все частные критерии оптимальности являются одинаково важными. Ранг агента s_i, $i \in [1:|S|]$ в его текущем положении X_i обозначаем r_i. В методе *NDS* используется простейшее из правил вычисления рангов, имеющее следующий вид (рисунок 7).

1) Выбираем среди всех агентов популяции недоминируемых, присваиваем им ранг, равный единице, и исключаем из дальнейшего рассмотрения.

2) Среди оставшихся агентов выбираем недоминируемых, присваиваем им ранг, равный двум, и исключаем из дальнейшего рассмотрения. И так далее до исчерпания популяции.

Ранг индивида легко использовать для вычисления его приспособленности, например, по формуле вида

$$\varphi(X_i) = \frac{1}{1+r_i}, \quad i \in [1:|S|].$$

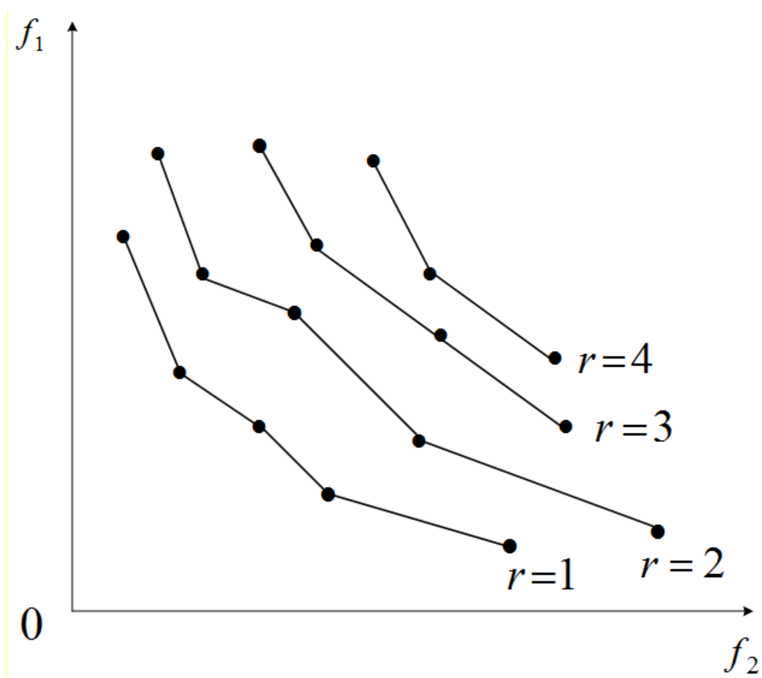

Рисунок 7 – К определению понятия ранга агента: $|F| = 2$

Предтечей метода недоминируемой сортировки можно, вероятно, считать *метод MOGA* □(*Multi-Objective Genetic Alg*orithm), предложенный Фонсека (*C. M. Fonseca*) и

Флемингом (*P. J. Fleming*) в 1993 г. [10]. В методе предполагается, что ранг агента равен числу агентов в популяции, которые его доминирует. Заслуживает также упоминания *Парето генетический алгоритм с нишеванием* (*Niched-Pareto Genetic Algorithm*, *NPGA*), предложенный Хорном (*J. Horn*) с соавторами в 1994 г. [20].

Как отмечалось выше, важнейшим требованием к методам и алгоритмам Парето-аппроксимации является требование обеспечения равномерности покрытия множества и фронта Парето. Для оценки равномерности покрытия может быть использована величина, называемая *разреженностью* (*scarcity*), имеющая смысл минимального расстояния между решениями, принадлежащими Парето-аппроксимации. Указанное расстояние может быть измерено с помощью различных метрик, например, с помощью известного манхеттеновского расстояния (*Manhattan distance*). Развитием метода недоминируемой сортировки является метод, в котором при формировании архивов A_X, A_F отвергают агентов, расстояние которых до других агентов в архиве не превышает заданной величины. Такой модифицированный метод положен в основу генетического алгоритма Парето-аппроксимации *NGSA*-II (*Non-Dominated Sorting Genetic Algorithm* II) [21].

5.2. Метод Парето-силы

Иное правило ранжирования агентов популяции используется в *методе Парето силы* (*Pareto strength*). Метод исходит из того, что сила агента s_i, $i \in [1:|S|]$ равна числу особей, которых доминирует по Парето данный агент. Определенную указанным способом силу агентов можно использовать для вычисления их приспособленности. Однако такое определение пригодности обладает рядом недостатков [14]. Поэтому в вычислительной практике используют альтернативное понятие *слабости* (*weakness*) агента, равной числу особей, его доминирующих. Точнее говоря, используют вариант слабости агента, называемый в работе [14] его *хилостью* (*wimpiness*) w_i и равный суммарной силе всех особей, доминирующих данного агента. Фитнесс-функция при этом имеет вид

$$\varphi_i = 1 / (1 + w_i), \quad i \in [1:|S|].$$

Метод Парето силы использует известный популяционный алгоритм Парето-аппроксимации *SPEA* (*Strength Pareto Evolutionary Algorithm*) [21]. Развитием алгоритма *SPEA* является алгоритм *SPEA*-2, который подобно алгоритму *NGSA*-II использует оценки равномерности Парето-аппроксимации [23].

Упрощенным вариантом алгоритма *SPEA*-2 можно считать алгоритм *PAES* (*Pareto Achieved Evolution Strategy*), предложенный Ноулзом (*J. Knowles*) и Корн (*D. Corne*) в 1999 г. [24]. Рангом r_i агента s_i в алгоритме *PAES* называется число архивных решений, доминируемых решением X_i; $i \in [1:|S|]$. Схема алгоритма *PAES* для агента s_i имеет следующий вид.

1) По законам используемого популяционного алгоритма (в оригинале – генетического алгоритма) перемещаем агента s_i из его текущего положения X_i в новое положение X'_i.

2) Если одно из решений X_i и X_i' доминирует другое решение, то лучшее из них объявляем *кандидатом*. В противном случае кандидатом объявляем то из решений X_i, X_i', ранг которого выше.

3) Если решение-кандидат доминируется некоторым из архивных решений, то кандидата отвергаем. В противном случае кандидата записываем в архив, но только в том случае, если он находится в той части области поиска, в которой на текущий момент обнаружено мало решений. При записи кандидата в архив все доминируемые им решения из архива удаляем.

4) Если условие окончания итераций не выполнено, возвращаемся к шагу 1.

5.3. Другие методы на основе ранжирования

*Метод средневзвешенного ранж*ирования (*Weighted Average Ranking, WAR*) предложили Бентли (*P. J. Bentley*) и Вэйкфилд (*J. P. Wakefield*) в 1996 г. [25]. Пусть $v_k = v(f_k)$ - относительная важность частного критерия f_k, $k \in [1:|F|]$. Метод *WAR* предполагает вычисление текущего ранга агента s_i, $i \in [1:|S|]$ по следующей схеме.

1) Для каждого из критериев f_k формируем список L_k, содержащий величины $f_k(X_i)$, упорядоченные по их возрастанию. $k \in [1:|F|]$

2) Ранг агента s_i полагаем равным величине

$$r_i = \sum_{k=1}^{|F|} \left(v_k \; n_k(X_i) \right), \quad i \in [1:|S|], \tag{7}$$

где $n_k(X_i)$ - порядковый номер агента s_i в списке L_k.

Метод суммы взвешенных оценок (*Sum of Weighted Ratios, SWR*) можно считать развитием метода *WAR* [25]. При вычислении ранга агентов метод использует свертку вида (7), в которой вместо относительной важности частных критериев v_k используются их особым образом нормализованные значения. Нормализация основана на использовании наихудших и наилучших текущих значений критериев

$$f^{worst}(X_i) = \max_{k \in [1:|F|]} f_k(X_i), \quad f^{best}(X_i) = \min_{k \in [1:|F|]} f_k(X_i)$$

и определяется формулой

$$\bar{f}_k(X_i) = \frac{f_k(X_i) - f^{worst}(X_i)}{f^{best}(X_i) - f^{worst}(X_i)}, \quad k \in [1:|F|], \quad i \in [1:|S|].$$

Метод суммы взвешенных глобальных оценок (*Sum of Weighted Global Ratios, SWGR*) [25] близок к предыдущему методу с тем отличием, что наихудшие и наилучшие значения критериев определяют на основе всей предыстории поиска, т.е. на основе всех итераций $\tau \in [1:t]$:

$$f^{worst}(X_i) = \max_{\tau \in [1:t]} \max_{l \in [1:|F|]} f_l(X_i^\tau); \quad f^{best}(X_i) = \min_{\tau \in [1:t]} \min_{l \in [1:|F|]} f_l(X_l^\tau); \quad i \in [1:|S|].$$

Метод максимального взвешенного ранжирования (*Weighted Maximum Ranking*, *WMR*) [25] основан на идее рассмотренного выше метода *VEGA*. Положим, что все критерии нормализованы и заданы их относительные важности v_k, $k \in [1:|F|]$. Схему метода определяет следующая последовательно шагов.

1) Аналогично методу *WAR* формируем списки L_k, $k \in [1:|F|]$ и определяем величины $n_k(X_i)$, $i \in [1:|S|]$.

2) Вычисляем ранги каждого из $|S|$ агентов по всем $|F|$ критериям оптимальности

$$r_k(X_i) = \frac{v_k}{n_k(X_i)}, \quad i \in [1:|S|], \quad k \in [1:|F|].$$

3) В качестве итогового ранга агента s_i в его текущем положении X_i принимаем величину

$$r_i = \max_{k \in [1:|F|]} r_k(X_i), \quad i \in [1:|S|].$$

6. Прочие методы

В данном разделе рассматриваем сигма-метод, методы композитных точек, гиперкубов, динамических соседей, а также метод хищник-жертва.

6.1. Сигма-метод

Сигма-метод (*sigma method*) предложили Мостагхим (*S. Mostaghim*) и Тич (*J. Teich*) в 2003 г. [26].

Для двухкритериальной задачи ($|F| = 2$) сигма-параметром точки X_i или соответствующей ей точки $F_i = (f_{i,1}, f_{i,2})$, называется величина

$$\sigma_i = \sigma(F_i) = \frac{f_{i,1}^2 - f_{i,2}^2}{f_{i,1}^2 + f_{i,2}^2}, \quad i \in [1:|S|].$$

Легко видеть, что точки F_i, имеющие одно и то же значение сигма-параметра, лежат на одной прямой, проходящей через начало системы координат $0 f_1 f_2$. Определение сигма-параметра иллюстрирует рисунок 8, на котором одинаковые значения сигма-параметра имеют пары точек $(F_1, F_2), (F_3, F_4), (F_5, F_6)$.

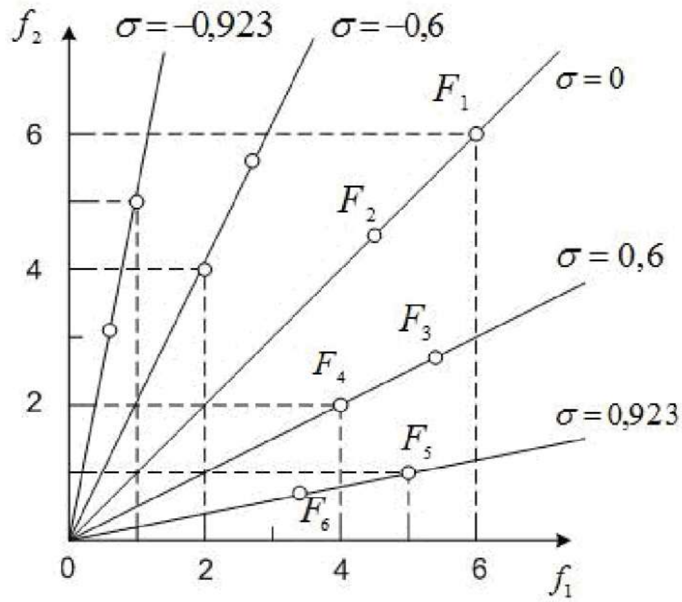

Рисунок 8 - К определению сигма-параметра: $|F| = 2$

В случае, когда размерность критериального пространства больше двух ($|F| > 2$) сигма-метод использует $|F|$-мерный вектор сигма-параметров $\sigma = (\sigma_1, \sigma_2, \ldots, \sigma_{|F|})$. Для агента s_i этот вектор определяет выражение

$$\sigma_i = \sigma(F_i) = \begin{pmatrix} f_{i,1}^2 - f_{i,2}^2 \\ f_{i,2}^2 - f_{i,3}^2 \\ \ldots \\ f_{i,|F|}^2 - f_{i,1}^2 \end{pmatrix} \cdot (f_{i,1}^2 + f_{i,2}^2 + \cdots + f_{i,|F|}^2), \quad i \in [1:|S|].$$

В сигма-методе для точки X_i, $i \in [1:|S|]$ лучшая архивная точка X_j^A, $j \in [1:|A|]$ определяется по следующей схеме.

1) Вычисляем значение сигма-параметра точки $F(X_i)$ - величину $\sigma_i = \sigma(F_i)$.

2) Находим в архиве A_F точку F_j^A, у которой значение сигма-параметра наиболее близко к величине σ_i:

$$\min_{l \in [1:|A|]} \| \sigma_l - \sigma_i \| \to \left\| \sigma_j - \sigma_i \right\|. \tag{8}$$

Здесь $\| * \|$ - символ евклидовой нормы.

3) В качестве лучшей для точки X_i принимаем точку X_j^A такую, что $F(X_j^A) = F_j^A$.

В данном случае в качестве метрики близости точек F_i, F_j^A используется евклидова норма в сигма-пространстве $\{\sigma\}$.

Утверждение 2. Предположим, что для всех архивных частиц $X_j^A,\ j \in [1:|A|]$ вектор сигма-параметров $\sigma = \sigma_1, \sigma_2, \ldots, \sigma_{|F|}$ уже вычислен. Тогда отыскание для точки X_i лучшей архивной точки X_j^A сигма-методом требует $O(|F||A|)$ арифметических операций.

Справедливость утверждения вытекает из того факта, что для отыскания точки, которая является лучшей для точки X_i, требуется $|A|$ раз вычислить значение нормы вида (8) и для каждого такого вычисления необходимо $O(|F|)$ арифметических операций•

Для двухкритериальной задачи изложенную схему сигма-метода иллюстрирует рисунок 9. Стрелками на рисунке для каждой из точек F_i показаны «притягивающие» точки F_j^A.

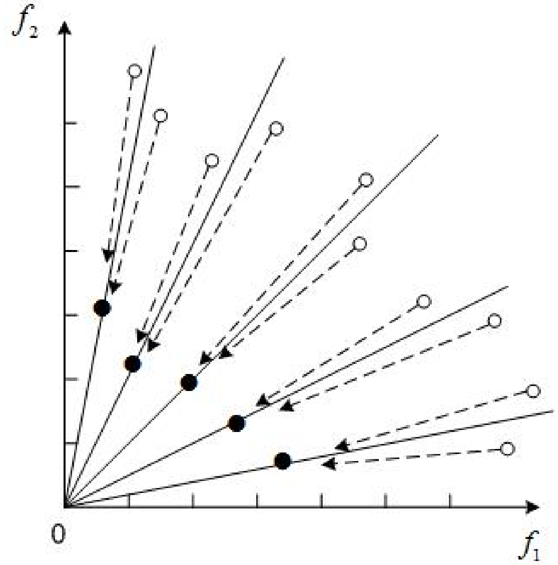

Рисунок 9 - Схема сигма-метода для двухкритериальной задачи:

● - архивные точки F_j^A; ○ - точки F_i

6.2. Метод композитных точек

Метод композитных точек предложили Филдсенд (*J.E. Fieldsend*) и Синх (*S. Singh*) в 2002 г. [27]. Метод использует архив A_F недоминируемых точек в виде, так называемого дерева доминирования (*dominated tree*), представляющего собой список *композитных точек* (*composite points*) $C_j = (c_{j,1}, c_{j,2}, \ldots, c_{j,|F|})$, $j = 1, 2, \ldots$. Для двухкритериальной задачи пример дерева доминирования приведен на рисунке 10.

Координаты первой композитной точки C_1 определяем по следующей схеме.

1) В качестве первой координаты $c_{1,1}$ этой композитной точки принимаем первую

координату той архивной точки $F^A_{j_1}$, $j_1 \in [1:|A|]$, у которой значение этой координаты наибольшее, т.е. полагаем

$$c_{1,1} = \max_{j \in [1:|A|]} f^A_{j,1} = f^A_{j_1,1}.$$

Исключаем точку $F^A_{j_1}$ из дальнейшего рассмотрения.

2) Вторую координату $c_{1,2}$ точки C_1 полагаем равной второй координате той из оставшихся архивных точек F^A_j, у которой эта координата имеет наибольшее значение, т.е. принимаем, что

$$c_{1,2} = \max_{j \in [1:|A|]} f^A_{j,2} = f^A_{j_2,2} \quad j \neq j_1.$$

Точку $F^A_{j_2}$ исключаем из дальнейшего рассмотрения.

3) Если $|F| > 2$, то поступаем аналогичным образом для всех остальных координат, так что

$$c_{1,|F|} = \max_{j \in [1:|A|]} f^A_{j,|F|} = f^A_{j_{|F|},|F|}, \;\; j \neq j_1, \;\; j \neq j_2, \ldots, j \neq j_{|F|-1}.$$

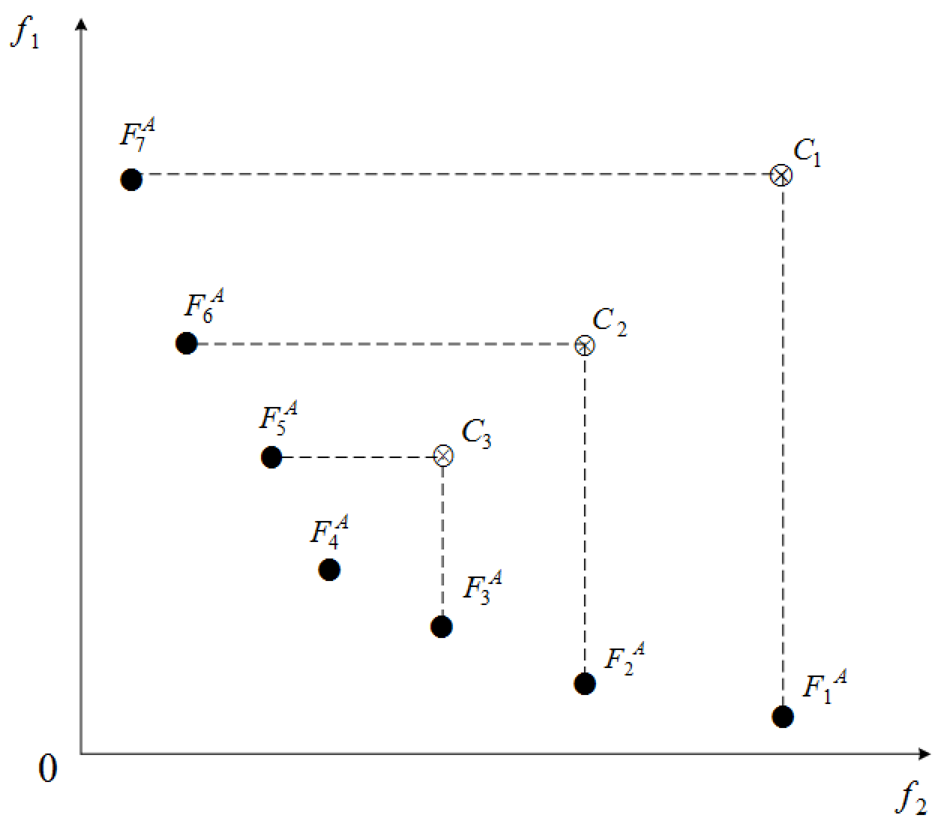

Рис. 10. Дерево доминирования для двухкритериальной задачи:

• – архивные точки $F_1^A - F_7^A$; $\otimes$ – композитные точки C_1, C_2, C_3

4) По аналогичной схеме вычисляем координаты второй композитной точки C_2 и так далее до исчерпания всех точек архива A_F.

Рассмотрим схему алгоритма быстрого построения дерева доминирования. Положим, что сформированы списки $L_1, L_2, \ldots, L_{|F|}$ из $|A|$ элементов каждый такие, что в списке L_k содержатся точки архива A_F, отсортированные в порядке убывания по критерию оптимальности $f_k(X)$; $k \in [1:|F|]$. При использовании известного алгоритма быстрой сортировки построение таких списков требует, в худшем случае, $O(|A|^2)$ операций сравнения. Элементы списка L_k обозначим $L_{k,j}$, $k \in [1:|F|]$, $j \in [1:|A|]$.

1) В качестве координаты $c_{1,k}$ точки C_1 принимаем k-ю координату элемента $L_{k,1}$ и исключаем этот элемент из всех списков.

2) В качестве координаты $c_{2,k}$ точки C_2 принимаем k-ю координату элемента $L_{k,2}$ и исключаем этот элемент из всех списков.

3) И так далее до исчерпания всех списков L_k; $k \in [1:|F|]$.

Легко видеть, что архиву A_F соответствует общее число композитных точек, равное $|C| = \left\lfloor \frac{|A|}{|F|} \right\rfloor$, где $\lfloor * \rfloor$ - символ ближайшего целого меньшего.

Архивные точки $F_{j_1}^A, F_{j_2}^A, \ldots, F_{j_{|F|}}^A$, соответствующие композитной точке C_j, назовем вершинами точки C_j; $j \in [1:|C|]$.

После того как дерево доминирования построено, лучшую точку для точки X_i определяем по следующему правилу.

1) В списке композитных точек находим точку C_j такую, что имеет место хотя бы одно из неравенств

$$c_{j,k} \le f_{i,k} < c_{j+1,j}, \quad j \in [1:(|C|-1)], \ k \in [1:|F|].$$

2) Из числа вершин $F_{j_1}^A, F_{j_2}^A, \ldots, F_{j_{|F|}}^A$ композитной точки C_j находим точку $F_{j_p}^A$, $p \in [1:|F|]$, которая строго доминирует точку F_i, т.е. такую, что $F_{j_p}^A \triangleright F_i$.

3) В качестве лучшей для точки X_i принимаем точку $X_{j_p}^A$, соответствующую вершине $F_{j_p}^A$.

4) Если среди вершин $F_{j_1}^A, F_{j_2}^A, \ldots, F_{j_{|F|}}^A$ имеется более одной, которая строго доминирует точку F_i, то из них случайным образом выбираем вершину $F_{j_p}^A$ и в качестве

лучшей для точки X_i принимаем соответствующую точку $X^A_{j_p}$.

Рассмотренное правило определения лучшей точки иллюстрирует рисунок 11 , на котором стрелками на рисунке показаны соответствующие лучшие точки.

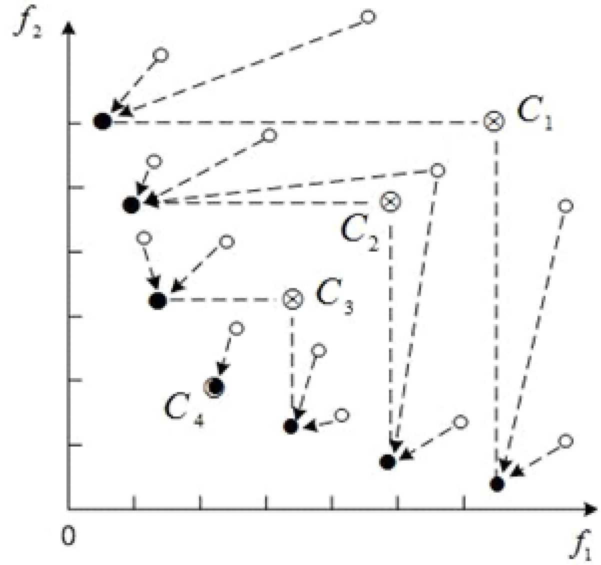

Рисунок 11 – К определению лучшей точки в методе композитных точек:
● – точки архива A_F; ⊗ – композитные точки; ○ - точки F_i

Утверждение 3. Предположим, что дерево доминирования предварительно тем или иным образом построено. Тогда поиск в архиве A_X точки, которая является лучшей для точки X_i, требует $O(|C|\cdot|F|^2)$ сравнений.

Справедливость утверждения вытекает из того факта, что в изложенной схеме определения лучшей точки поиск композитной точки C_j требует $O(|C|\cdot|F|)$, а выбор точки $F^A_{j_p}$ - не более $O(|F|)$ сравнений•

6.3. Метод гиперкубов

В основе *метода гиперкубов*, предложенного Коэлло (*C.A. Coello*) и Лечунга (*M.S. Lechunga*) в 2002 г. [28], лежит процедура покрытия области достижимости D_F некоторой совокупностью гиперкубов и процедура назначения каждому из этих гиперкубов коэффициента пригодности. Рассмотрим данные процедуры.

Положим частные критерии оптимальности $f_1(X),\ f_2(X),\ldots,\ f_{|F|}(X)$ нормированными, так что $f_k(X)\in[0;1],\ k\in[1:|F|],\ \forall X\in D_X$.

Искомые гиперкубы строим путем разделения $|F|$-мерного единичного гиперкуба $H^0=[0;1]\times[0;1]\times\ldots\times[0;1]$ на $n^{|F|}$ гиперкубов с длиной ребра, равной $\dfrac{1}{n}$ (рисунок 12), где натуральное число n - свободный параметр метода.

Пусть некоторый гиперкуб $H_j^1,\ j \in [1:n^{|F|}]$ включает в себя m_j архивных точек. Пригодность этого гиперкуба принимаем равной величине $\varphi_j = \dfrac{10}{m_j}$. Если выбранный гиперкуб вообще не содержит точек архива, то происходит расширение области поиска, т.е. вводится в рассмотрение гиперкуб, который включает в себя гиперкуб H_j^1. Таким образом, пригодность каждого из построенных гиперкубов полагаем обратно пропорциональной числу архивных точек, которые находятся в нем, и гиперкуб, содержащий меньшее число этих точек, оказывается более пригодным (рисунок 12). Такое определение пригодности направлено на обеспечение более равномерной Парето-аппроксимации.

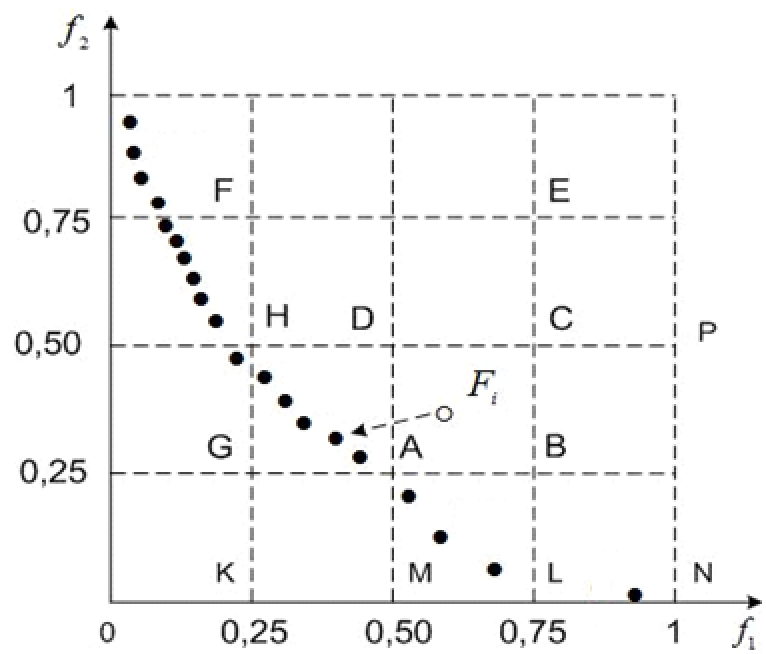

Рисунок 12 – Покрытие области D_F гиперкубами для двухкритериальной задачи: $n = 4$; • – архивные точки F_j^A

Если по рассмотренному правилу множество достижимости D_F покрыто гиперкубами и для каждого из гиперкубов определены их пригодности, то отыскание в архиве A_X точки, которая является лучшей для точки X_i, производится по следующей схеме.

1) Если в гиперкубе H_j^1, в котором находится точка X_i, имеются архивные точки, то случайным образом с равной вероятностью выбираем одну из этих точек $F_{j_i}^A$.

2) В противном случае выполняем следующие действия:

- расширяем зону поиска - рассматриваем все возможные гиперкубы H_j^2 с длиной ребра равной $\frac{2}{n}$, которые включают в себя гиперкуб H_j^1;

- определяем пригодность гиперкубов H_j^2, как сумму пригодностей входящих в них гиперкубов H_j^1;

- из числа всех найденных гиперкубов H_j^2 по правилу рулетки выбираем один из гиперкубов H_{j*}^2;

- случайным образом с равной вероятностью выбираем одну из точек $F_{j_i}^A$ этого гиперкуба.

3) В качестве искомой лучшей точки принимаем соответствующую точку $F_{j_i}^A$.

Приведенную схему выбора лучшей точки иллюстрирует рисунок 12. Точка F_i на рисунке находится в гиперкубе $ABCD$. В этом гиперкубе нет ни одной архивной точки. Поэтому в рассмотрение вводим гиперкубы $BEFG, CHKL, DMNP$. Пригодности этих гиперкубов равны 2; 2,3; 0,4 соответственно. По правилу рулетки выбираем, например, гиперкуб $BEFG$. В этом гиперкубе с равной вероятностью выбираем одну из архивных точек, например, точку, на которую на рисунке указывает стрелка.

6.4. Метод динамических соседей

Метод динамических соседей предложили Хью (*X. Hu*) и Эберхарт (*R. Eberhart*) в 2002 г. [29]. Схему метода можно представить в следующем виде.

1) Фиксируем все критерии оптимальности, кроме критерия $f_1(X)$ - полагаем этот критерий *динамическим*.

2) Вычисляем в пространстве критериев $f_2, f_3,..., f_{|F|}$ расстояния (например, евклидовы) от точки F_i, $i \in [1:|S|]$ до всех архивных точек F_j^A, $j \in [1:|A|]$.

3) Выбираем на этой основе m ближайших архивных соседей точки F_i, где m - свободный параметр метода.

4) Из числа выделенных точек выбираем лучшую по критерию $f_1(X)$.

5) На следующей итерации полагаем динамическим критерий $f_2(X)$, а все остальные критерии фиксируем. И так далее.

Авторы метода отмечают, что остается открытым вопрос об оптимальной величине числа m.

6.5. Метод хищник-жертва

Метод хищник-жертва (*predator-prey*) предложил Лауманс (*M. Laumanns*) с соавторами в 1998 г. [30]. Рассматриваем оригинальный метод Лауманса и его основные модификации последних лет. Полагаем, что областью допустимых значений вектора варьируемых параметров является гиперпараллелепипед

$$D_X = (X \mid x_i^- \le x_i \le x_i^+,\ i \in [1:|X|]).$$

Каждая из жертв представляет собой один вектор решений, а каждый хищник - один критерий оптимальности. Популяция эволюционирует на тороидальной сетке (рисунок 13), в узлах которой случайным образом инициализируем жертв и хищников. Каждый хищник рассматривает все жертвы в своей окрестности и удаляет жертву с худшим значением соответствующего критерия оптимальности. Затем в той же окрестности выбираем и модифицируем случайную жертву. Измененное решение помещаем на место удаленной жертвы. Хищника случайным образом перемещаем в один из соседних с ним узлов сетки. Процесс итерационно повторяем до выполнения условий окончания.

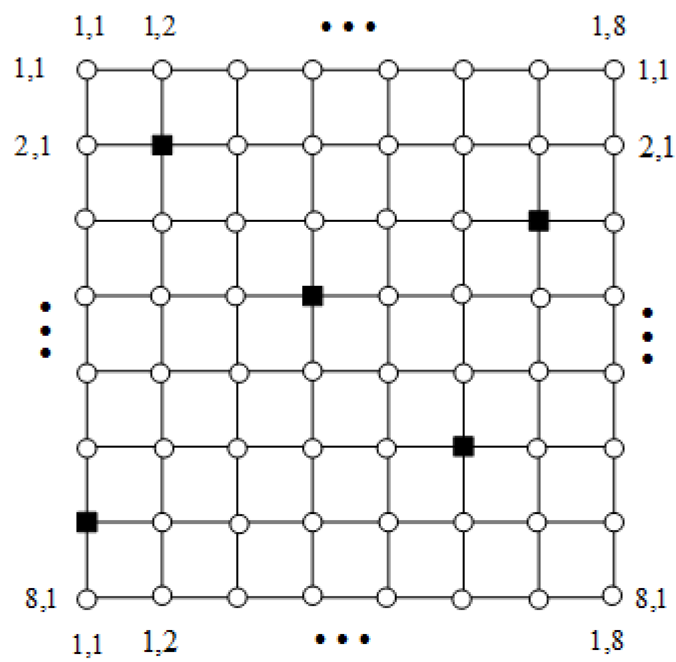

Рисунок 13 – Пример размещения хищников (квадратики) и жертв (кружки) на тороидальной сетке

Приведем схему метода в более формализованном виде.

1) В гиперпараллелепипеде D_X случайным образом инициализируем популяцию жертв – решения $X_i,\ i \in [1:|S|]$.

2) Ставим в соответствие жертвам вершины ненаправленного связного графа в виде тороидальной решетки.

3) Случайным образом помещаем хищников в вершины указанного графа.

4) Ставим в соответствие каждому из хищников один из критериев оптимальности $f_k(X),\ k \in [1:|F|]$ таким образом, чтобы каждый из критериев был представлен, по меньшей мере, одним хищником.

5) Вычисляем значения соответствующего критерия для всех жертв в окрестности каждого из хищников и выбираем худшие жертвы.

6) Полагаем, что выбранные жертвы поглощены хищниками, и удаляем их из популяции.

7) Взамен каждого из удаленных агентов создаем новую жертву путем изменения случайно выбранной жертвы в окрестности удаленного агента.

8) Случайным образом перемещаем хищников из их текущих положений в соседние вершины графа.

9) Если условие окончания итераций не выполнено, возвращаемся к шагу 5.

Экспериментально показано, что с ростом числа поколений популяция жертв стремится к фронту Парето, т.е. представляет собой искомую Парето-аппроксимацию.

Недостатком рассмотренного канонического метода хищник-жертва является возможная потеря лучших (ближайших к множеству Парето решений).

Деб (*K. Deb*) в 2001 г. предложил улучшения метода [31], суть которых заключается в следующем:

- хищникам ставят в соответствие не частные критерии оптимальности, а векторы их весов;

- вместо удаленной жертвы новую жертву создают путем модификации не случайного, но лучшего агента в окрестности удаленной жертвы;

- хищника перемещают не случайным образом, а в направлении позиции лучшей жертвы в его окрестности.

Исследования авторов метода показывают, что он обеспечивает по сравнению с каноническим методом более высокую скорость сходимости, однако, как и канонический метод, может терять найденные лучшие решения.

В работе [32] рассмотрена модификация канонического метода, предложенная Ли (*X. Li*), в которой жертв и хищников помещают не во все узлы графа, представленного на рисунке 13. Жертвы и хищники выполняют случайные перемещения в направлении соседних пустых узлов графа, причем хищники движутся быстрее жертв. При каждом перемещении жертвы создается ее потомок. Вычислительные эксперименты показали недостаточно высокую скорость сходимости соответствующего алгоритма.

В той же работе [32] Дебом (*K. Deb*) и Рао (*U. Rao*) предложена глубокая модификация канонического метода, основу которой составляют следующие три механизма:

- механизм сохранения лучших решений;

- механизм рекомбинации решений;

- механизм получения равномерной Парето-аппроксимации.

Механизм сохранения лучших решений (элитизм) предполагает сравнения приспособленности худших жертв с приспособленностью созданных агентов (потомков). Меру приспособленности агентов строим на основе Парето доминирования. Если данный потомок слабо доминирует всех жертв популяции (лучше их, по крайней мере, по одному из частных критериев оптимальности), то заменяем этим потомком соответствующую худшую жертву. В противном случае потомка объявляем неподходящим, худшую жертву оставляем в популяции, а хищник для обеспечения разнообразия популяции случайным образом перемещаем в направлении худшей жертвы.

Механизм рекомбинации решений строим на основе операторов скрещивания и мутации агентов (так что, в целом, метод оказывается вариантом генетического алгоритма). В окрестности худшей жертвы создаем двух потомков путем скрещивания лучшего и второго лучшего решений. Затем случайным образом выбираем одного из этих потомков и выполняем его мутацию. Если полученный таким образом потомок оказывается хуже худшей жертвы, то по рассмотренной схеме создаем нового потомка и опять сравниваем его приспособленность с приспособленностью худшей жертвы. Процесс повторяем не более некоторого фиксированного числа раз.

Механизм обеспечения равномерности покрытия. Полагаем, что каждая жертва имеет *область влияния* (*influencing region*) в виде гиперкуба в критериальном пространстве, в центре которого находится данная жертва. Потомка полагаем не подходящим, если он создан в области влияния любой из существующих жертв. Заметим, что аналогичный механизм используется в методе ε-доминирования [33]. Размеры области влияния назначаем, исходя из требуемой равномерности покрытия.

Для того чтобы предотвратить ситуацию, когда хищники полностью уничтожают популяцию, используем следующий прием. Число перемещений t^P, которые выполняют хищники перед тем, как жертвы могут сделать свои перемещения, объявляем динамической переменной, определяемой формулой

$$t^P = \left\lfloor \frac{\left|S^p\right| - \left|S_*^p\right|}{\left|S^P\right|} \right\rfloor,$$

где $\lfloor \bullet \rfloor$ - символ ближайшего меньшего, $\left|S^p\right|$, $\left|S_p^*\right|$- текущее и желаемое числа жертв, $\left|S^P\right|$ - текущее число хищников. В результате при уменьшении числа жертв уменьшается величина t^P, т.е. хищники начинают двигаться медленнее, шансы жертв на выживание повышаются, и популяция жертв расширяется. При этом хищники становятся более быстрыми, и их популяция начинает сокращаться. И так далее.

Авторы рассматриваемой модификации метода выполнили широкое экспериментальное исследование его эффективности на ряде трудных задач известного тестового набора двух- и трехцелевых задач многокритериальной оптимизации [31]. Показано, что алгоритм обеспечивает высокие скорость сходимости и равномерность покрытия фронта-Парето.

Заключение

Многообразие методов построения Парето-аппроксимации указывает на актуальность данной задачи. Представляются перспективными следующие направления развития этих методов.

Вычислительная сложность частных критериев оптимальности, вообще говоря, различна в различных частях области варьируемых параметров D_X. Рассмотренные и иные известные методы Парето-аппроксимации не учитывают этого обстоятельства. Поэтому актуальной, на наш взгляд, является разработка адаптивных и самоадаптивных методов Парето-аппроксимации, учитывающих данный факт.

Практически значимые задачи многокритериальной оптимизации имеют высокую вычислительную сложность и требуют для своего решения использования параллельных вычислительных систем. Поэтому необходима разработка параллельных методов Парето-аппроксимации, ориентированных на различные классы таких систем (кластерные системы, графические процессорные устройства и т.д.) [34].

Важным с практической точки зрения является класс задач Парето-аппроксимации, в которых требуется построить аппроксимацию не всего фронта Парето, но лишь той части его, которая ближе всего к заданной пользователем «предпочтительной» точке пространства критериев. Поэтому представляется актуальной задача разработки методов решения задач данного класса. Известна модификация метода хищник-жертва, предназначенная для решения таких задач [32]. Можно также считать, что метод идеальной точки является частным случаем этих методов.

Задачу Парето-аппроксимации можно считать одной из широкого класса задач приближенного построения границ множеств различной природы. Поэтому значительный интерес представляет адаптация рассмотренных методов и разработка новых методов, предназначенных для решения задач указанного класса. Пример такого подхода демонстрирует работа [35], в которой один из методов Парето-аппроксимации использован для приближенного построения границы области достижимости многосекционного манипулятора параллельной кинематики типа хобот.

Отметим, наконец, перспективность гибридизации рассмотренных и иных методов Парето-аппроксимации.

Авторы благодарят Бушневского А.В. и Савелова А.С. за помощь в подготовке некоторых материалов для данной работы.

Литература

1. Подиновский В.В., Ногин В.Д. Парето-оптимальные решения многокритериальных задач.- М.: Физматлит, 2007.- 256 с.

2. Ногин В.Д. Принятие решений в многокритериальной среде: количественный подход.- М.: Физматлит, 2005.- 176 с.

3. Ларичев О.И. Теория и методы принятия решений: Учебник для ВУЗов.- М.: Университетская книга, Логос, 2006.- 392 с.

4. Курейчик В.М. Генетические алгоритмы и их применение.- Таганрог: Изд-во Таганрогского РТУ, 2002. -244 с.

5. Ashlock D. Evolutionary Computation for Modeling and Optimization.- Springer, 2006.- 571 pp.

6. Guliashki V., Toshev H., Korsemov Ch. Survey of Evolutionary Algorithms Used in Multiobjective Optimization // Problems of Engineering Cybernetics and Robotics, 2009, Vol. 60, pp. 42 – 54.

7. Zitzler E., Deb K., Thiele L. Comparison of Multiobjective Evolutionary Algorithms: Empirical Results // Evolutionary Computation, 2000, Vol. 8(2), pp. 173-195.

8. Карпенко А. П., Селиверстов Е. Ю. Глобальная оптимизация методом роя частиц. Обзор // Информационные технологии, 2010, №2, с. 25-34.

9. Карпенко А. П., Селиверстов Е. Ю. Обзор методов роя частиц для задачи глобальной оптимизации // Наука и образование: электронное научно-техническое издание, 2009, №3. (http://technomag.edu.ru/doc/116072.html).

10. Fonseca C. M., Fleming P. J. Genetic Algorithms for Multiobjective Optimization: Formulation, Discussion and Generalization / Proc. of the 5th International Conference on Genetic Algorithms, San Mateo, California, 1993, pp. 416-423.

11. Zitzler E., Thiele L., Marco Laumanns M., Fonseca C. M., da Fonseca V. G. Performance Assessment of Multiobjective Optimizers: An Analysis and Review // IEEE Transactions of Evolutionary Computation, 2003, Vol. 7(2), pp. 117-132.

12. Knowles J.D., Corne D.W. On Metrics for Comparing Non-Dominated Sets // Proc. of the 2002 IEEE Congress on Evolutionary Computation (CEC '02), New Jersey, Piscataway, May 2002, IEEE Service Center, 2002, Vol. 1, pp. 711-716.

13. Гуменникова А. В. Адаптивные поисковые алгоритмы для решения сложных задач многокритериальной оптимизации. Диссертация на соискание ученой степени к. т. н.- Красноярск, 2006, 129 с.

14. Luke S. Essentials of Metaheuristics. A Set of Undergraduate Lecture Notes. Department of Computer Science George Mason University, Online Version 1.3 February, 2012. (http://cs.gmu.edu/~sean/book/metaheuristics/Essentials.pdf).

15. Соболь И.М., Статников Р.Б. Выбор оптимальных параметров в задачах со многими критериями: Учебное пособие для ВУЗов.- М.: Дрофа, 2006.- 175 с.

16. Schaffer J.D. Multiple Objective Optimization with Vector Evaluated Genetic Algorithms / In: Genetic Algorithms and Their Applications. Proceedings of the First International Conference on Genetic Algorithms, 1985, Lawrence Erlbaum, pp. 93-100.

17. Моор Д.А., Мухлисуллина Д. Т. Анализ эффективности различных сверток критериев оптимальности в задаче многокритериальной оптимизации // Наука и образование: электронное научно-техническое издание, 2010, №4. (http://technomag.edu.ru/doc/141623.html).

18. Ryu J.-H., Kim S., Wan H. Pareto front approximation with adaptive weighted sum method in multiobjective simulation optimization / Proceedings of the 2009 Winter Simulation Conference (WSC), 2009, Austin, pp. 623-633. (http://www.informs-sim.org/wsc09papers/060.pdf).

19. Srinivas N., Deb K. Multiobjective optimization using nondominated sorting in genetic algorithms // Evolutionary Computation, 1994, vol. 2, p. 221–248.

20. Horn J., Nafpliotis N., Goldberg D.E. A Niched Pareto Genetic Algorithm for Multiobjective Optimization / Proc. of the First IEEE Conference on Evolutionary Computation, IEEE World Congress on Computational Intelligence, New Jersey, 1994, Vol. 1, pp. 82-87.

21. Deb K., Pratap A., Agarwal S., Meyarivan T. A fast and elitist multiobjective genetic algorithm: NSGA-II // IEEE Transactions on Evolutionary Computation, 2002, Vol. 6, Issue 2, pp. 182 – 197.

22. Zitzler E., Thiele L. Multiobjective evolutionary algorithms: A comparative case study and the strength Pareto approach, IEEE Transactions on Evolutionary Computation, 1999, Vol. 3(4), pp. 257–271.

23. Zitzler E., Laumanns M., Thiele L. SPEA2: Improving the Strength Pareto Evolutionary Algorithm for Multiobjective Optimization / In K.C. Giannakoglou and al., editors. Evolutionary Methods for Design, Optimization and Control with Application to Industrial Problems (EUROGEN 2001), International Center for Numerical Methods in Engineering (CIMNE), 2002, pp. 95-100.

24. Knowles J., Corne D. The Pareto achieved evolution strategy: A new baseline algorithm for Pareto multiobjective optimization / Proceedings of the Congress on Evolutionary Computation (CEC99), 1999, Washington, Vol. 1, pp. 98–105.

25. Bentley P. J., Wakefield J. P. An Analysis of Multiobjective Optimization within Genetic Algorithms / Technical Report ENGPJB96, University of Huddersfield, UK, 1996. P. 19.

26. Mostaghim S., Teich J. Strategies for Finding Good Local Guides in Multi-objective Particle Swarm Optimization (MOPSO) / In: Swarm Intelligence Symposium, 2003. SIS '03. Proceedings of the 2003 IEEE, pp. 26 – 33.

27. Fieldsend J.E., Singh S. A multi-objective algorithm based upon particle swarm optimization, an efficient data structure and turbulence (2002). (http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.19.6959).

28. Coello C.A., Lechuga M.S. MOPSO: A proposal for multiple objective particle swarm optimization / In IEEE Proceedings, World Congress on Evolutionary Computation (CEC2002), 2002, pp. 1051-1056.

29. Hu X., Eberhart R. Multiobjective Optimization Using Dynamic Neighborhood Particle Swarm Optimization / In: Evolutionary Computation, 2002. CEC '02. Proceedings of the 2002 Congress on, Honolulu, HI, USA, 12 – 17 May 2002, pp. 1677 – 1681.

30. Laumanns M., Rudolph G., Schwefel H.P. A spatial predator-prey approach to multi-objective optimization: A preliminary study / In Proceedings of the Parallel Problem Solving from Nature, 1998, Vol. V, pp. 241-249.

31. Deb K. Multi-objective optimization using evolutionary algorithms.- Chichester, UK: Wiley, 2001, P. 518.

32. Deb K., Rao U. B. Investigating Predator-Prey Algorithms for Multi-Objective Optimization / Kanpur Genetic Algorithms Laboratory (KanGAL), Department of Mechanical Engineering Indian Institute of Technology Kanpur, India, KanGAL Report Number 2005010 (http://www.iitk.ac.in/kangal/).

33. Deb K., Mohan M., Mishra S. Evaluating the ε-domination based multi-objective evolutionary algorithm for a quick computation of Pareto-optimal solutions // Evolutionary Computation Journal, 2005, Vol. 13(4), pp. 501-525.

34. Антух А.Э., Карпенко А.П., Семенихин А.С., Хасанова Р.В. Построение множества Парето методом роя частиц на графических процессорах архитектуры CUDA // Научный сервис в сети Интернет: суперкомпьютерные центры и задачи: Труды Международной суперкомпьютерной конференции (20-25 сентября 2010 г., г. Новороссийск). М.: Изд-во МГУ, 2010. С. 274-280.

35. Карпенко А.П., Семенихин А.С., Червяцова М.Н. Метод приближенного построения границы области достижимости многосекционного манипулятора типа «хобот» // Наука и образование: электронное научно-техническое издание, 2011, № 1. (http://technomag.edu.ru/doc/165078.html).

electronic scientific and technical periodical
SCIENCE and EDUCATION
EL № FS 77 - 30569. №0421100025. ISSN 1994-0408

Review: population methods of Pareto set approximation in multi-objective optimization problem

77-30569/363023

 # 04, April 2012

Karpenko A.,P., Semenikhin A., S., Mitina E.V.

Bauman Moscow State Technical University
apkarpenko@mail.ru
saspost@yandex.ru
mitinakaterina@gmail.com

This authors present a review of numerical methods of approximate Pareto set generation in the multi-objective optimization problem. The following methods are discussed: "naive" methods, switching objective functions methods, methods of objective functions aggregation, methods based on ranking of population agents, and other methods. All cases referred to methods involving the use of genetic or swarm algorithms, such as particle swarm optimization algorithm.

Publications with keywords: multiobjective optimization, Pareto set, Pareto front, population methods
Publications with words: multiobjective optimization, Pareto set, Pareto front, population methods

References

1. Podinovskii V.V., Nogin V.D. *Pareto-optimal'nye resheniia mnogokriterial'nykh zadach* [Pareto-optimal solutions of multicriterion problems]. Moscow, Fizmatlit, 2007. 256 p.

2. Nogin V.D. *Priniatie reshenii v mnogokriterial'noi srede: kolichestvennyi podkhod* [Decision-making in multicriterion environment: a quantitative approach]. Moscow, Fizmatlit, 2005. 176 p.

3. Larichev O.I. *Teoriia i metody priniatiia reshenii* [Theory and methods of decision making]. Moscow, Universitetskaia kniga, Logos, 2006. 392 p.

4. Kureichik V.M. *Geneticheskie algoritmy i ikh primenenie* [Genetic algorithms and their applications]. Taganrog, TRSU Publ., 2002. 244 p.

5. Ashlock D. *Evolutionary Computation for Modeling and Optimization*. Springer, 2006. 571 p.

6. Guliashki V., Toshev H., Korsemov Ch. Survey of Evolutionary Algorithms Used in Multiobjective Optimization. *Problems of Engineering Cybernetics and Robotics*, 2009, vol. 60, pp. 42 – 54.

7. Zitzler E., Deb K., Thiele L. Comparison of Multiobjective Evolutionary Algorithms: Empirical Results. *Evolutionary Computation (journal)*, 2000, vol. 8, no. 2, pp. 173-195.

8. Karpenko A. P., Seliverstov E. Iu. Global'naia optimizatsiia metodom roia chastits. Obzor [Global optimization of method of particle swarm. Overview]. *Informatsionnye tekhnologii*, 2010, no. 2, pp. 25-34.

9. Karpenko A. P., Seliverstov E. Iu. Obzor metodov roia chastits dlia zadachi global'noi optimizatsii [Review of the particle swarm optimization method (PSO) for a global optimization problem]. *Nauka i obrazovanie: nauchnoe izdanie MGTU im. N.E. Baumana* [Science and Education: scientific periodical of the Bauman MSTU], 2009, no. 3. Available at: http://technomag.edu.ru/doc/116072.html, accessed 4.05.2012.

10. Fonseca C. M., Fleming P. J. Genetic Algorithms for Multiobjective Optimization: Formulation, Discussion and Generalization. *Proc. of the 5th International Conference on Genetic Algorithms*, San Mateo, California, 1993, pp. 416-423.

11. Zitzler E., Thiele L., Marco Laumanns M., Fonseca C. M., da Fonseca V. G. Performance Assessment of Multiobjective Optimizers: An Analysis and Review. *IEEE Transactions of Evolutionary Computation*, 2003, Vol. 7 (2), pp. 117-132.

12. Knowles J.D., Corne D.W. On Metrics for Comparing Non-Dominated Sets. *Proc. of the IEEE Congress on Evolutionary Computation (CEC '02)*, Vol. 1, New Jersey, Piscataway, May 2002, IEEE Service Center, 2002, pp. 711 – 716.

13. Gumennikova A. V. *Adaptivnye poiskovye algoritmy dlia resheniia slozhnykh zadach mnogokriterial'noi optimizatsii. Kand. diss.* [Adaptive search algorithms for solving complex problems of multicriteria optimization. Cand. diss.]. Krasnoiarsk, 2006. 129 p.

14. Luke S. *Essentials of Metaheuristics. A Set of Undergraduate Lecture Notes*. Department of Computer Science George Mason University, Online Version 1.3 February, 2012. Available at: http://cs.gmu.edu/~sean/book/metaheuristics/Essentials.pdf.

15. Sobol' I.M., Statnikov R.B. *Vybor optimal'nykh parametrov v zadachakh so mnogimi kriteriiami* [The choice of optimal parameters in problems with many criteria]. Moscow, Drofa, 2006. 175 p.

16. Schaffer J.D. Multiple Objective Optimization with Vector Evaluated Genetic Algorithms. *Proc. of the First International Conference on Genetic Algorithms and Their Applications*, 1985, Lawrence Erlbaum, pp. 93-100.

17. Moor D.A., Mukhlisullina D. T. Analiz effektivnosti razlichnykh svertok kriteriev optimal'nosti v zadache mnogokriterial'noi optimizatsii [Analysis of the effectiveness of different scalar convolutions in multiobjective optimization problem]. *Nauka i obrazovanie: nauchnoe izdanie MGTU im. N.E. Baumana* [Science and Education: scientific periodical of the Bauman

MSTU], 2010, no. 4. Available at: http://technomag.edu.ru/doc/141623.html, accessed 4.05.2012.

18. Ryu J.-H., Kim S., Wan H. Pareto front approximation with adaptive weighted sum method in multiobjective simulation optimization. *Proc. of the 2009 Winter Simulation Conference (WSC)*, 2009, Austin, pp. 623-633. Available at: http://www.informs-sim.org/wsc09papers/060.pdf.

19. Srinivas N., Deb K. Multiobjective optimization using nondominated sorting in genetic algorithms. *Evolutionary Computation (journal)*, 1994, vol. 2, p. 221–248.

20. Horn J., Nafpliotis N., Goldberg D.E. A Niched Pareto Genetic Algorithm for Multiobjective Optimization. *Proc. of the First IEEE Conference on Evolutionary Computation, IEEE World Congress on Computational Intelligence*, New Jersey, Piscataway, June 1994, IEEE Service Center, Vol. 1, pp. 82-87.

21. Deb K., Pratap A., Agarwal S., Meyarivan T. A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE Transactions on Evolutionary Computation*, 2002, vol. 6, no. 2, pp. 182 – 197.

22. Zitzler E., Thiele L. Multiobjective evolutionary algorithms: A comparative case study and the strength Pareto approach. *IEEE Transactions on Evolutionary Computation*, 1999, vol. 3, no. 4, pp. 257–271.

23. Zitzler E., Laumanns M., Thiele L. SPEA2: Improving the Strength Pareto Evolutionary Algorithm for Multiobjective Optimization. *Evolutionary Methods for Design, Optimization and Control with Application to Industrial Problems (EUROGEN 2001)*, International Center for Numerical Methods in Engineering (CIMNE), 2002, pp. 95‐100.

24. Knowles J., Corne D. The Pareto achieved evolution strategy: A new baseline algorithm for Pareto multiobjective optimization. *Proc. of the Congress on Evolutionary Computation (CEC99)*, Washington, 1999, Vol. 1, pp. 98–105.

25. Bentley P. J., Wakefield J. P. *An Analysis of Multiobjective Optimization within Genetic Algorithms*. Technical Report ENGPJB96, University of Huddersfield, UK, 1996. 19 p.

26. Mostaghim S., Teich J. Strategies for Finding Good Local Guides in Multi-objective Particle Swarm Optimization (MOPSO). *Proc. of the Swarm Intelligence Symposium (SIS '03)*, IEEE, 2003, pp. 26 – 33.

27. Fieldsend J.E., Singh S. *A multi-objective algorithm based upon particle swarm optimization, an efficient data structure and turbulence*. 2002. Available at:

http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.19.6959.

28. Coello C.A., Lechuga M.S. MOPSO: A proposal for multiple objective particle swarm optimization. *Proc. of the World Congress on Evolutionary Computation (CEC2002)*, IEEE Service Center, 2002, pp. 1051-1056.

29. Hu X., Eberhart R. Multiobjective Optimization Using Dynamic Neighborhood Particle Swarm Optimization. *Proc. of the Congress on Evolutionary Computation (CEC '02)*, May 2002, IEEE Service Center, pp. 1677 – 1681.

30. Laumanns M., Rudolph G., Schwefel H.P. A spatial predator-prey approach to multi-objective optimization: A preliminary study. *Fifth International Conference on Parallel Problem Solving from Nature (PPSN-V)*, Berlin, Germany, pp. 241-249.

31. Deb K. *Multi-objective optimization using evolutionary algorithms*. Chichester, UK, Wiley, 2001. 518 p.

32. Deb K., Rao U. B. *Investigating Predator-Prey Algorithms for Multi-Objective Optimization*. KanGAL Report Number 2005010. Kanpur Genetic Algorithms Laboratory (KanGAL), Department of Mechanical Engineering Indian Institute of Technology Kanpur, India. Available at: http://www.iitk.ac.in/kangal/.

33. Deb K., Mohan M., Mishra S. Evaluating the -domination based multi-objective evolutionary algorithm for a quick computation of Pareto-optimal solutions. *Evolutionary Computation Journal*, 2005, vol. 13, no. 4, pp. 501-525.

34. Antukh A.E., Karpenko A.P., Semenikhin A.S., Khasanova R.V. Postroenie mnozhestva Pareto metodom roia chastits na graficheskikh protsessorakh arkhitektury CUDA [The construction of the Pareto set using method of a swarm of particles on the GPUs CUDA architecture]. *Nauchnyi servis v seti Internet: superkomp'iuternye tsentry i zadachi: Trudy Mezhdunarodnoi superkomp'iuternoi konferentsii* (20-25 sentiabria 2010 g., g. Novorossiisk) [Scientific services on the Internet: supercomputing centers and Objectives: Proc. of the International Supercomputer Conference (20-25 September 2010, Novorossiysk)]. Moscow, Bauman MSTU Publ., 2010, pp. 274-280.

35. Karpenko A.P., Semenikhin A.S., Cherviatsova M.N. Metod priblizhennogo postroeniia granitsy oblasti dostizhimosti mnogosektsionnogo manipuliatora tipa «khobot» [Approximation of a set of attainability for trunk multisectional robot-manipulator]. *Nauka i obrazovanie: nauchnoe izdanie MGTU im. N.E. Baumana* [Science and Education: scientific periodical of the Bauman MSTU], 2011, no. 1. Available at: http://technomag.edu.ru/doc/165078.html, accessed 4.05.2012.