Решение частично целочисленной задачи выпуклой нелинейной оптимизации методом последовательных аппроксимаций многогранниками

Максим Гончаров
mmaxgon@yandex.ru
январь 2022

1 Введение

Задачи частично целочисленной выпуклой нелинейной оптимизации часто возникают в ритейле, например, для нахождения оптимальных условий промо-акций, включающих как непрерывные переменные решения – глубину скидок, так и дискретные, например, выбор промо-механик и типов выкладки, а также логические переменные решения, такие как флаг выбора товара в промо или логические переменные, описывающие правила совместного вхождения нескольких товаров в промо-акцию. При этом часть ограничений и функция цели могут рассчитываться нелинейными функциями, что переводит проблему в класс MINLP (Mixed Integer Nonlinear Programming).

В настоящее время доступен широкий набор бесплатных солверов, предназначенных для решения нелинейных частично целочисленных задач. Например, входящие в проект COIN-OR Mindtpy, Bonmin, Couenne, SHOT или APOPT, входящий в проект Gekko, и другие. Существует также большое количество коммерческих MINLP-солверов. Однако общим для них существенным ограничением является необходимость символьной формулировки задачи, т.е. ограничения и функция цели должны быть описаны в виде алгебраических выражений. Этот подход неприемлем в случае, если функция цели или часть нелинейных ограничений доступны как внешние функции, рассчитываемые сложными прогнозными моделями. Например, в случае максимизации спроса в зависимости от условий и контекста предложения, численно рассчитываемого как интеграл по плотности вероятности спроса, восстановленного глубокой нейронной сетью, целевую функцию невозможно представить алгебраически.

В этой связи возникает потребность в MINLP-солверах, позволяющих использовать функциональность существующих символьных оптимизационных движков, но расширяющих область их применения на функции, заданные в виде «чёрных ящиков».

Далее в статье подробно описывается алгоритм решения MINLP-задач при помощи итеративного вызова одного из символьных солверов для подзадачи с последующим её уточнением. В этом подходе функции, описываемые как «чёрные ящики», на каждой итерации всё более точно представляются в виде аппроксимирующих многогранников. Сходимость аппроксимаций к исходным функциям гарантируется выпуклостью этих функций.

Доказывается сходимость описанного алгоритма к оптимальному значению на уровне функции цели и на уровне подпоследовательности решений. Обсуждается набор эвристик, ускоряющих сходимость и уменьшающих размерность задачи.

2 Постановка задачи

Обозначим непрерывные и целочисленные переменные решения как $x_1, \ldots, x_k \in \mathbb{R}$ и $y_1, \ldots, y_m \in \mathbb{Z}$ соответственно. Для простоты мы будем обозначать далее переменные решения в виде $z = (x, y) = (x_1, \ldots, x_k, y_1, \ldots, y_m) \in \mathbb{R}^k \times \mathbb{Z}^m$.

Мы предполагаем, что на переменные решения снизу и сверху наложены ограничения. Эти ограничения задают компактную область определения переменных решения. Компактность нам нужна для последующего доказательства сходимости на уровне подпоследовательности решений. Т.е. существует компакт K, такой что $z = (x_1, \ldots, x_k, y_1, \ldots, y_m) \in K \subset \mathbb{R}^k \times \mathbb{Z}^m$.

Обозначим через H область, заданную системой равенств и неравенств, которые можно описать в символьном виде и с типом, поддерживаемым выбранным вспомогательным солвером (например, для линейного солвера CBC, это могут быть только линейные выражения, для SHOT – выражения, описывающие выпуклые нелинейные функции). Т.е. $H := \{z \in \mathbb{R}^k \times \mathbb{Z}^m \mid A(z) \ \leq b, C(z) = d\}$, где A и C – непрерывные векторные функции от переменных решения, представляемые в виде алгебраических выражений, а b и d – векторы ограничений.

Обозначим через G область, заданную системой неравенств, которые нельзя описать в символьном виде, т.е. $G := \{z \in \mathbb{R}^k \times \mathbb{Z}^m \mid g(z) \ \leq 0\}$, где $g(z) = (g_1(z), \ldots g_n(z)) \colon \mathbb{R}^{k+m} \to \mathbb{R}^n$ – выпуклая, непрерывно дифференцируемая, n-компонентная векторная функция, которая представляет собой «чёрный ящик» для вспомогательного солвера и не может быть задана напрямую в решаемой им задаче. Дифференцируемость функции g необходима, так как мы должны будем иметь возможность строить в каждой точке промежуточного решения касательные к некоторым компонентам этой функции. Выпуклость необходима для того, чтобы множество, задаваемое пересечением касательных к различным точкам границы G, последовательно аппроксимировало множество G.

Из определения дифференцируемости и выпуклости g непосредственно следует:

$$g_i(z) \geq g_i(z_0) + \langle \nabla g_i(z_0), z - z_0 \rangle, \qquad (1)$$

$\forall z, z_0 \in \mathbb{R}^k \times \mathbb{R}^m, \forall i \in \{1, \ldots, n\}$

[Выпуклость $\Rightarrow g_i\big(z_0 + t(z - z_0)\big) \leq g_i(z_0) + t(g_i(z) - g_i(z_0)), \forall t \in [0,1]$

Дифференцируемость $\Rightarrow \langle \nabla g_i(z_0), z - z_0 \rangle = \lim\limits_{t \to 0} \frac{g_i(z_0 + t(z - z_0)) - g_i(z_0)}{t} \leq g_i(z) - g_i(z_0) \Rightarrow (1)$],

Где $\langle \cdot, \cdot \rangle$ — скалярное произведение в $\mathbb{R}^{k+m}$.

Неравенство (1) будет использовано при исследовании свойств аппроксимирующих множеств, заданных касательными.

Так как функции A, C , g предполагаются непрерывными, то области H и G, которые они определяют через равенства и неравенства – замкнутые, а следовательно допустимая область $X := K \cap H \cap G$ как пересечение замкнутых множеств внутри компакта K является компактной. Мы предполагаем, что X - непустое множество. Определим также компактное множество $X_0 := K \cap H$, которое будет начальным приближением множества X.

Минимизируемая функция цели $f \colon \mathbb{R}^{k+m} \to \mathbb{R}$ также предполагается «чёрным ящиком», т.е. её нельзя определить символьно. (Если это не так, то задача упрощается: мы задаём эту функцию в виде алгебраического выражения в вспомогательный солвер, и дальнейшая задача будет заключаться в нахождении её минимума при помощи последовательной аппроксимации множества G). Мы также накладываем на fтребования по выпуклости и непрерывной дифференцируемости. По аналогии с (1) получаем:

$f(z) \geq f(z_0) + \langle \nabla f(z_0), z - z_0 \rangle,$ (2)
$\forall z, z_0 \in \mathbb{R}^k \times \mathbb{R}^m$

Так как целевая функция f непрерывна, а допустимое множество X компактно и непусто, то у задачи
$f(z) \rightarrow min,\ z \in X$ (3)
есть оптимальное решение $z^* = argmin\{f(z) | z \in X\}$

Для конечного множества точек $\{z_0, z_1, \ldots, z_t\}$ из X_0 и номера $j \in \{1, \ldots, t\}$ определим линейную аппроксимацию функции f в точке z_{j-1}:
$f_j(z) := f\left(z_{j-1}\right) + \langle \nabla f\left(z_{j-1}\right), z - z_{j-1} \rangle$ (4)
и кусочно-линейную аппроксимацию функции f в точках $z_0, \ldots, z_{j-1}$:
$F_j(z) := max\{f_1(z), \ldots f_j(z)\}$ (5)
Т.е. f_j – касательная под функцией f в точке z_{j-1}, а F_j – функция, составленная из касательных к f в точках $z_0, \ldots, z_{j-1}$, аппроксимирующая f снизу.

Из (2) следует:
$f_j(z) \leq f(z), \forall z, \forall j$ (6)
$F_j(z) = max\{f_1(z), \ldots f_j(z)\} \leq f(z), \forall z, \forall j$
Из определения F_j как максимума конечного числа функций следует монотонность по номеру индекса:
$F_j(z) = max\{f_1(z), \ldots f_j(z)\} \leq max\{f_1(z), \ldots f_j(z), f_{j+1}(z), \ldots, f_i(z)\} = F_i(z)\ \forall z, \forall j \leq i$ (7)

Далее, для каждого $j \in \{1, \ldots, t\}$ определим множество индексов функции g, в которых ограничения $g\left(z_{j-1}\right) \leq 0$ нарушаются:
$I_j := \{p \in \{1, \ldots, n\}\ |\ g_p(z_{j-1}) > 0\}$ (8)
Для индексов нарушенных ограничений функции g в точке z_{j-1} построим замкнутое множество
$G_j := \{z \in \mathbb{R}^k \times \mathbb{Z}^m\ |\ g_p(z_{j-1}) + \langle \nabla g_p\left(z_{j-1}\right), z - z_{j-1} \rangle\ \leq 0, \forall p \in I_j\}$ (9)
Если ограничения функции g в точке z_{j-1} нигде не нарушаются, т.е. если $I_j = \emptyset$, то $G_j := \mathbb{R}^k \times \mathbb{Z}^m$.
Определим область $X_j := K \cap H \bigcap_{i=1}^{j} G_i = X_0 \bigcap_{i=1}^{j} G_i$ (10)
Т.е. X_j получается последовательным наложением ограничений линеаризованных функций g на множество X_0.
Из определения X_j следует, что на каждой итерации это множество может только уменьшиться (или остаться таким же, если все ограничения в соответствующей точке соблюдены), т.е.:
$X_0 \supset X_1 \supset \cdots \supset X_t$
Но на каждой итерации справедливо $X \subset X_j, \forall j$, а следовательно, получаем:
$X_0 \supset \cdots \ldots \supset X_t \supset X$ (11)
Таким образом, мы можем рассматривать X_j как последовательность линейных аппроксимаций допустимого множества X сверху.
[Действительно, на каждой итерации j+1 для получения множества X_{j+1} мы пересекаем X_j с множеством G_{j+1}, поэтому $X_j \supset X_{j+1}$. Теперь докажем, что $X \subset X_j$. Предположим, что существует $z \in X \backslash X_j$. Тогда $z \in X = K \cap H \cap G \subset K \cap H;\ z \notin X_j = K \cap H \bigcap_{i=1}^{j} G_i \Rightarrow z \notin \bigcap_{i=1}^{j} G_i \Rightarrow \exists i \in \{1, \ldots, j\}: z \notin G_i\ \Rightarrow I_i \neq \emptyset$ и $\exists p \in I_i$: $g_p(z_{i-1}) + \langle \nabla g_p(z_{i-1}), z - z_{i-1} \rangle\ > 0$. Но из (1) следует $g_p(z) \geq g_p(z_{i-1}) + \langle \nabla g_p(z_{i-1}), z - z_{i-1} \rangle$, а из $z \in X = K \cap H \cap G \subset G$ следует $g_p(z) \leq 0$. Из трёх последних неравенств получаем: $0 \geq g_p(z) \geq g_p(z_j) + \langle \nabla g_p\left(z_j\right), z - z_j \rangle > 0$. Мы пришли к противоречию 0>0, следовательно $X \backslash X_j = \emptyset$, а значит $X \subset X_j$.]

Так как каждое множество X_j компактно (пересечение замкнутых и компактного) и непустое (содержит X), то целевая функция f принимает на нём минимум:

$$z_j^* := argmin\{f(z) \mid z \in X_j\} \tag{12}$$

Так как X_j уменьшаются, то минимумы f на X_j увеличиваются и с учётом (11) ограничены сверху минимумом f на X:

$$f(z_0^*) \le f(z_1^*) \le \ldots \le f(z_p^*) \le f(z^*) \tag{13}$$

Теперь мы готовы сформулировать алгоритм и доказать его сходимость.

3 Алгоритм

0. На нулевом шаге положим $j := 0$ и с формулируем задачу для вспомогательного солвера (далее – ВС) без функции цели: найти $z_0 \in X_0 = K \cap H$. Так как непустое компактное множество X_0 определяется через символьные равенства и неравенства, обрабатываемые ВС, то задача решается. Увеличим номер шага $j := 1$ и переходим к пункту 1.

1. На шаге $j > 0$для уже найденной на прошлом шаге точки $z_{j-1} \in X_{j-1}$ выбираем множество компонентов $p \in \{1, \ldots, n\}$, для которых ограничения $g_{\boldsymbol{p}}(z_{j-1}) \le 0$ не соблюдаются, т.е. определим множество $I_j := \{p \in \{1, \ldots, n\} \mid \ g_p(z_{j-1}) > 0\}$. Если $I_j \ne \emptyset$, то определим замкнутое множество, заданное *линейными* неравенствами $G_j := \{z \in \mathbb{R}^k \times \mathbb{Z}^m \mid g_p(z_{j-1}) + \langle \nabla g_p(z_{j-1}), z - z_{j-1} \rangle \ \le 0, \forall p \in I_j\}$, а если $I_j = \emptyset$, то $G_j := \mathbb{R}^k \times \mathbb{Z}^m$. Добавим эти ограничения к допустимому множеству задачи ВС прошлого шага: $X_j := X_{j-1} \cap G_j = X_0 \bigcap_{i=1}^{j} G_i$. Таким образом, X_j получается добавлением к X_{j-1}конечного множества линейных неравенств, а следовательно, может обрабатываться ВС. Из (11) следует, что $X_0 \supset X_1 \supset \ldots \supset X_{j-1} \supset X_j \supset X$, т.е. X_j – непустое компактное множество, включающее допустимое множество X исходной задачи и уменьшающее допустимое множество задачи ВС прошлого шага.

2. Определим линейную функцию $f_j(z) := f(z_{j-1}) + \langle \nabla f(z_{j-1}), z - z_{j-1} \rangle$, аппроксимирующую функцию цели исходной задачи f снизу в виде касательной в точке z_{j-1}.

   Определим кусочно-линейную функцию
   $F_j(z) := max\{f_1(z), \ldots f_j(z)\}$, аппроксимирующую f снизу в виде максимума касательных в точках $z_1, \ldots, z_{j-1}$.

3. Решим вспомогательную задачу шага j:

   $$F_j(z) \to min, \tag{14}$$
   $$z \in X_j$$

   Как уже отмечалось, область X_j задана равенствами и неравенствами, обрабатываемыми ВС, так как к первоначальному набору символьных равенств и неравенств, определяющих множество X_0, на последующих итерациях добавляются только линейные неравенства, описывающие множество G_j.

   Функция $F_j(z)$ – кусочно-линейная и обрабатывается символьным выпуклым вспомогательным солвером. Если вспомогательный солвер может работать только с линейной задачей, то (14) можно эквивалентно переформулировать в задачу частично-целочисленного линейного программирования, введя дополнительную переменную решения μ:

$\mu \to min,$ (15)
$z \in X_j,$
$f_1(z) \le \mu, \dots f_j(z) \le \mu$

Задачи (14) и (15) действительно эквиваленты, так как минимизация максимума функций эквивалента минимизации их общей верхней границы.

Таким образом, мы получаем задачу, обрабатываемую ВС. Найденное решение вспомогательной задачи на шаге j – следующая точка конструируемой последовательности z_j. Т.е. $z_j := argmin\{F_j(z) \mid z \in X_j\}$ (16)

Для минимальной точки $z_j^* := argmin\{f(z) \mid z \in X_j\}$ исходной функции цели f на том же множестве X_j справедливо:
$F_j(z_j) \le F_j(z_j^*) \le f(z_j^*) \le f(z_j)$ (17)
[Действительно: из неравенства (6) следует $F_j(z_j^*) = max\{f_1(z_j^*), \dots f_j(z_j^*)\} \le f(z_j^*)$. Так как z_j^* минимизирует f на X_j, то $f(z_j^*) \le f(z_j)$, а так как z_j минимизирует F_j на X_j, то $F_j(z_j) \le F_j(z_j^*)$. Из трёх последних неравенств получаем: $F_j(z_j) \le F_j(z_j^*) \le f(z_j^*) \le f(z_j)$.]

4. Увеличим номер шага $j := j + 1$ и переходим к пункту 1.

Таким образом, мы получаем бесконечную последовательность точек $z_j \in X_j$ на последовательности сужающихся множеств X_j, каждое из которых является внешней линейной аппроксимацией допустимого множество исходной задачи X. Каждая полученная точка является решением вспомогательной оптимизационной задачи $z_j = argmin\{F_j(z) \mid z \in X_j\}$ с функцией цели, являющейся кусочно-линейной аппроксимацией снизу исходной функции цели f.

Верхней границей оценки оптимального решения $f(x^*)$ на итерации t служит (уменьшающийся при увеличении числа итераций) минимум функции цели по всем точкам последовательности, удовлетворяющим исходным ограничениям, т.е.
$Upper(t) := \min\{f(z_j) \big| j \le t, z_j \in X\} \ge f(z^*)$.

Нижней границей решения является (увеличивающееся с увеличением числа итераций) значение функции кусочно-линейной аппроксимации исходной функции цели снизу, т.е.
$Lower(t) := F_t(z_t)$
[Действительно, для $i \le j \le t$ из (11) следует, что $X_j \subset X_i$, а так как $z_i \in X_i$ – минимум F_i на X_i, а $z_j \in X_j \subset X_i$, то $F_i(z_i) \le F_i(z_j)$. С другой стороны, из (7) следует $F_i(z_j) \le F_j(z_j)$. В свою очередь, $z_j \in X_j$ – минимум F_j на X_j, а так как из (11) следует $z^* \in X \subset X_j$, то $F_j(z_j) \le F_j(z^*)$. И, наконец, из (6) следует $F_j(z^*) \le f(z^*)$. Объединяя все неравенства вместе, получаем: $F_i(z_i) \le F_j(z_j) \le f(z^*)$. Таким образом, $F_i(z_i)$ – возрастающая с ростом номера итерации последовательность, ограниченная сверху оптимальным значением $f(z^*)$. Следовательно, $F_t(z_t)$ может служить оценкой $f(z^*)$ снизу.]

Имея на каждой итерации оценку оптимального значения сверху и снизу, алгоритм можно остановить, как только разница между этими оценками станет достаточно малой.

4 Сходимость

Так как $z_j \in X_j \subset X_0$, т.е. каждая точка z_jнаходится в компактном множестве X_0, то существует подпоследовательность z_{j_s}, сходящаяся к некоторой точке $\overline{z} \in X_0$, т.е. $\lim\limits_{s\to\infty} z_{j_s} = \overline{z}$.

Утверждение 1: Предельная точка любой сходящейся подпоследовательность z_j является допустимой, т.е. из $\lim\limits_{s\to\infty} z_{j_s} = \overline{z} \in X_0$ следует $\overline{z} \in X$.

Доказательство: Предположим, что это не так, т.е. $\overline{z} \in X_0 \backslash X$. Так как $X = X_0 \cap G$, то из этого следует $\overline{z} \notin G = \{z \mid g(z) \leq 0\}$. Следовательно, существует индекс $p \in \{1, \dots, n\}$ такой, что $g_p(\overline{z}) > 0$. Из сходимости z_{j_s}к $\overline{z}$ при $s \to \infty$ и непрерывности g_p следует существование номера $s_0 \in \mathbb{N}$ такого, что $g_p(z_{j_s}) > 0, \forall s \geq s_0$. При выполнении пункта 1 алгоритма, для каждого такого s на шаге $j_s + 1$ на множество X_{j_s+1} , а следовательно, и на все последующие множества поиска решений, накладывается ограничение $g_p(z_{j_s}) + \langle \nabla g_p(z_{j_s}), z - z_{j_s} \rangle \ \leq 0$. Т.е. все последующие $z_i, i > j_s$ должны этому ограничению удовлетворять, в том числе и следующий элемент сходящейся подпоследовательности $z_{j_{s+1}}$, т.е. $g_p(z_{j_s}) + \langle \nabla g_p(z_{j_s}), z_{j_{s+1}} - z_{j_s} \rangle \ \leq 0$. Переходя в последнем неравенстве к пределу $s \to \infty$, и учитывая $z_{j_{s+1}} - z_{j_s} \to 0, \nabla g_p(z_{j_s}) \to \nabla g_p(\overline{z})$ (непрерывная дифференцируемость g_p) и $g_p(z_{j_s}) \to g_p(\overline{z})$ (непрерывность g_p), получаем $g_p(\overline{z}) \leq 0$, что противоречит изначальному предположению $g_p(\overline{z}) > 0$. Из полученного противоречия следует, что $\overline{z} \in X$. Утверждение доказано.

Утверждение 2: Предельная точка любой сходящейся подпоследовательность z_j является оптимальной, т.е. из $\lim\limits_{s\to\infty} z_{j_s} = \overline{z} \in X_0$ следует $\overline{z} = argmin\{f(z) \mid z \in X\}$.

Доказательство: Из утверждения 1 следует, что $\overline{z}$ – допустимая точка, т.е. $\overline{z} \in X$. Обозначим z_j^* $:= argmin\{f(z) \mid z \in X_j\}$.

Так как $z_{j_s}, z_{j_s}^* \in X_{j_s}$и $z_{j_s}^*$ – минимум f на X_{j_s}, то

$$f(z_{j_s}) \geq f(z_{j_s}^*) \tag{18}$$

Учитывая неравенство (6): $F_j(z) = max\{f_1(z), \dots f_j(z)\} \leq f(z), \forall z, \forall j$, т.е. тот факт, что F_j при любом индексе j является аппроксимацией f снизу, получаем:

$$f(z_{j_s}^*) \geq F_{j_s}(z_{j_s}^*) \tag{19}$$

Так как по определению точка z_{j_s} минимизирует функцию F_{j_s} на множестве X_{j_s}, а $z_{j_s}^* \in X_{j_s}$, то получаем:

$$F_{j_s}(z_{j_s}^*) \geq F_{j_s}(z_{j_s}) \tag{20}$$

Далее из (7) следует $F_j(z) = max\{f_1(z), \dots f_j(z)\} \leq max\{f_1(z), \dots f_j(z), f_{j+1}(z), \dots, f_i(z)\} = F_i(z)$, $\forall z, \forall j \leq i$. Учитывая то, что для двух соседних индексов подпоследовательности j_{s-1}и j_s найдётся натуральное число $t > 0$, такое что $j_s = j_{s-1} + t \geq j_{s-1} + 1$, получаем $j_s \geq j_{s-1} + 1$ и неравенство (7) преобразуется в $F_{j_s}(z) \geq \ F_{j_{s-1}+1}(z), \forall z$. Получаем:

$$F_{j_s}(z_{j_s}) \geq \ F_{j_{s-1}+1}(z_{j_s}) \tag{21}$$

Из определения $F_{j_{s-1}+1}$ получаем $F_{j_{s-1}+1}(z) = max\{f_1(z), \dots, f_{j_{s-1}+1}(z)\} \geq f_{j_{s-1}+1}(z)$. Из определения $f_{j_{s-1}+1}$на 2-ом шаге алгоритма получаем: $f_{j_{s-1}+1}(z) = f(z_{j_{s-1}}) + \langle \nabla f(z_{j_{s-1}}), z - z_{j_{s-1}} \rangle$.
Подставляем z_{j_s}:

$$F_{j_{s-1}+1}(z_{j_s}) \geq f_{j_{s-1}+1}(z_{j_s}) = \ f(z_{j_{s-1}}) + \langle \nabla f(z_{j_{s-1}}), z_{j_s} - z_{j_{s-1}} \rangle \tag{22}$$

Объединяем неравенства (18)-(22):

$f(z_{j_s}) \geq f(z_{j_s}^*) \geq f\big(z_{j_{s-1}}\big) + \langle \nabla f\big(z_{j_{s-1}}\big), z_{j_s} - z_{j_{s-1}} \rangle$ (23)

Переходя к пределу $s \to \infty$, и учитывая, что $z_{j_s} - z_{j_{s-1}} \to 0, \nabla f\big(z_{j_{s-1}}\big) \to \nabla f(\overline{z})$, (непрерывная дифференцируемость f), получаем $\lim_{s\to\infty} \langle \nabla f\big(z_{j_{s-1}}\big), z_{j_s} - z_{j_{s-1}} \rangle = 0$, а учитывая, что $f(z_{j_s}), f\big(z_{j_{s-1}}\big) \to f(\overline{z})$ (непрерывность f), из (23) получаем:

$\lim_{s\to\infty} f(z_{j_s}^*) = f(\overline{z})$ (24)

Так как решение исходной задачи $z^* = argmin\{f(z) \mid z \in X\}$ минимизирует функцию f на множестве X, а $z_{j_s}^*$ минимизирует функцию f на бо**о**льшем множестве $X_{j_s} \supset X$, то $f(z_{j_s}^*) \leq f(z^*)$. Подставляя в (24), получаем: $f(\overline{z}) = \lim_{s\to\infty} f(z_{j_s}^*) \leq f(z^*)$. **(25)**

Так как согласно утверждению 1: $\overline{z} \in X$, а z^* минимизирует f на X, то : $f(z^*) \leq f(\overline{z})$. Вместе с (25) получаем: $f(\overline{z}) = f(z^*)$. Т.е. $\overline{z}$ тоже минимизирует f на X, а следовательно, является решением исходной задачи. Утверждение доказано.

Утверждение 3: Последовательность значений целевой функции от рассчитываемой алгоритмом последовательности точек z_j сходится к оптимальному значению целевой функции на допустимой области, т.е. $\lim_{j\to\infty} f(z_j) = f(z^*) = \min\{f(z) \mid z \in X\}$.

Доказательство: Предположим, что это не так. Тогда существует $\varepsilon > 0$, и такая подпоследовательность z_{j_s}, что $|f z_{j_s}) - f(z^*)| > \varepsilon, \forall s \in \mathbb{N}$ (26)

Так как все z_{j_s} лежат в компактном множестве X_0, то существует дальнейшая подпоследовательность $z_{j_{s_r}}$ и точка $\overline{z} \in X_0$, такие что $\lim_{r\to\infty} z_{j_{s_r}} = \overline{z}$. Из утверждений 1 и 2 следует, что $\overline{z}$ – оптимальная точка исходной задачи, т.е. $\overline{z} = argmin\{f(z) \mid z \in X\}$, а следовательно, $f(\overline{z}) = f(z^*)$. Из непрерывности f следует существование такого $r_0 \in \mathbb{N}$, что $|f(z_{j_{s_r}}) - f(\overline{z})| \leq \varepsilon, \forall r \geq r_0$. Следовательно, $|f(z_{j_{s_r}}) - f(z^*)| \leq \varepsilon, \forall r \geq r_0$, что противоречит (26). Таким образом, мы доказали, что $\lim_{j\to\infty} f(z_j) = f(z^*)$.

Из утверждений 1, 2 и 3 следует, что рассчитываемая алгоритмом последовательность точек z_j обладает следующими свойствами:

1. Функция цели от z_j сходится к минимальному значению функции цели на допустимом множестве.
2. Предел любой сходящейся подпоследовательности является оптимальным решением исходной задачи.

5 Эвристики

Ниже приведены изменения базового алгоритма, иногда значительно повышающие скорость его работы и уменьшающие требования к памяти. Эти изменения доказали свою эффективность на многочисленных практических примерах.

1. В пункте 1 алгоритма на каждой j-ой итерации в множество ограничений добавляются линейные ограничения вида $g_p(z_{j-1}) + \langle \nabla g_p\big(z_{j-1}\big), z - z_{j-1} \rangle \leq 0$ для *всех* индексов p, в которых в точке z_{j-1} нарушаются ограничения, т.е. для всех индексовp таких, что $g_p\big(z_{j-1}\big) > 0$. Практика показывает, что часто более эффективно добавлять только одно ограничения такого вида, например, соответствующее максимальному значению нарушения ограничений в точке. Т.е. выбирать такой p, для которого $g_p\big(z_{j-1}\big) = \max\{g_t\big(z_{j-1}\big) \mid t \in \{1, \dots, n\}: g_t\big(z_{j-1}\big) > 0\}$. В этом случае на каждом шаге создаётся меньше ограничений, что при большом числе итераций

может сильно ускорить расчёт каждой итерации и уменьшить используемый объём памяти. При этом добавление только одного ограничения достаточно, чтобы данная недопустимая точка была уже в дальнейшем исключена из допустимой области.

2. Если в задаче используется большое число непрерывных переменных решения, то полезно на каждой итерации после получения вспомогательного решения решать ещё одну оптимизационную задачу: зафиксировать целочисленные переменные (полученные ВС), задать все, в том числе нелинейные ограничения и нелинейную функцию цели в нелинейный непрерывный солвер, например, scipy.optimize или NLOPT, не требующий представления задачи в символьном виде, и решать задачу минимизации исходной целевой функции на исходном (нелинейном) допустимом множестве, но только на подмножестве непрерывных переменных. Таким образом, мы уточняем полученное решение только по непрерывным переменным, получая дополнительно лучшую верхнюю оценку оптимальной функции цели.

3. Практические эксперименты показали, что поиск допустимого решения можно существенно ускорить если добавлять линейные ограничения $g_p(z_{j-1}) + \langle \nabla g_p(z_{j-1}), z - z_{j-1} \rangle \ \leq 0$ не в виде касательных к точкам z_{j-1}, полученным на предыдущей итерации, для которых ограничения нарушаются, а для их проекций z'_{j-1} на допустимое множество X. В этом случае недопустимые точки тоже удаляются в дальнейшем из допустимой области вспомогательных задач X_j, но дополнительно благодаря тому, что касательные строятся в точках на границе исходного допустимого множества, такие ограничения в гораздо большей степени отсекают недопустимые фрагменты области из дальнейшего рассмотрения, что приводит к значительному уменьшению числа необходимых итераций. Задачу нахождения проекции недопустимой точки на исходную допустимую область можно решать с использованием нелинейного непрерывного солвера, в который подаются все исходные, в том числе нелинейные ограничения, а функцией цели служит евклидово расстояние от данной недопустимой точки до допустимого множества. Решение не соблюдает требования по целочисленности и служит только для более эффективного отсечения недопустимых фрагментов при помощи касательных к граничным точкам.

4. Имеет смысл удалять из задачи ограничения, добавленные на ранних шагах и представляющие собой касательные к недопустимым точкам, полученным вдали от текущих решений. Эти ограничения уже не участвуют в формировании последних решений задачи, но усложняют постановку вспомогательной задачи. Для этого проверяется факт нахождения нескольких последних решений близко друг от друга и вдали от решений начальных итераций. После чего соответствующие ограничения удаляются.

6 Литература

1. Dimitri P. Bertsekas, ”Convex Optimization Algorithms”, Athena Scientific, 2015
2. Andreas Lundella, Jan Kronqvistb, Tapio Westerlund “The Supporting Hyperplane Optimization Toolkit for Convex MINLP”, 2021