

Optimal Approximation of Non-linear Power Flow Problem*

Nina Vishnyakova

ninavishn@yandex.ru

Moscow Institute of Physics and Technology (State University)

This paper addresses Optimal Approximation of Non-linear Power Flow Problem. To solve it, the “trust region approach” method is used, which implies a consequent consideration of convex optimization problems. The obtained results are compared with results of well-known methods such as SDP Relaxation, SOC Relaxation, etc. (1). The rate of convergence of studied method for large dimensions problems with tens of thousands of nodes is investigated. The results of direct computations confirm a higher rate of convergence for the proposed method and a reduction in the computational resources, necessary for performing each iteration.

Index terms: *optimal power flow, non-convex optimization problem, trust region approach, convex restriction .*

1 Introduction

In this article we explore the problem of optimization of the workload in the energy network in order to reduce the cost of electricity production. This is a very important topic in the power industry (2). Energy flows are huge, so any results, leading to small improvements in the scale of the power grid, lead to a tremendous economic effect [YM: blue]“Energy flows are huge...” – try to rewrite the sentence. Take a look on the paper you cite; what is there motivation?.

The problem is called AC-Optimal Power Flow problem AC-OPF problem) [YM: We have two problems: AC-OPF and DC-OPF, the second one is a linear programming]. In general case this optimization problem with constraints, imposed by nonlinear Kirchhoff’s equations and the needs of consumers, has a convex functional and non-convex constraints. A large number of articles offer various solutions. Linear approximation method suggests linearizing the power flow constraints. Local optimization methods are about looking for a local optimum of the AC-OPF problem. And global optimization methods propose convexifying the constraints imposed by the Kirchhoff’s laws.

In this paper, we solve the OFP problem using the “trust region approach” method, which implies a consequent consideration of convex optimisation problems. We study the convergence of the method and the rate of convergence in relation to large dimension problems with tens of thousands of nodes.

The obtained results are compared with another methods: DC Approximation, LPAC Approximation, SDP Relaxation, SOC Relaxation, QC Relaxation. The results of direct computations confirm a fast rate of convergence for the proposed method and a reduction in the computational resources, necessary for performing each iteration.

[YM: Nina, Very good! Sorry for the very brief comments, I have a short stopover on my way to Europe. Few comments are below:

- References are missing. I have added a few (2–5), take a look please. In (2) you will find a proper setup and a motivation behind the problem;
- try to support each statement by a citation, e.g., the problem hardness, the problem motivation etc.

* Academic supervisor: Vadim V. Strijov. The problem author: Michael Chertkov. Consultant: Yury Maximov.

|

2 Power network structure

Consider a power network modeled by a directed graph $\mathbf{G}(\mathcal{N}, \mathcal{E})$ where buses are represented by nodes in \mathcal{N} and transmission lines by edges in $\mathcal{E} \subset \mathcal{N} \times \mathcal{N}$.

Вершины могут быть различных типов. А именно, генераторы и нагрузки. Каждый узел также может содержать шунтирующий элемент. Generator is modeled as a complex power injection at a specific bus. For generator i , the injection is:

$$s_g^i = p_g^i + jq_g^i. \quad (1)$$

A generator with a negative injection can be also used to model a dispatchable load.

Constant power loads are modeled as a specific quantity of real and reactive power consumed at a bus. For bus i , the load is

$$s_d^i = p_d^i + jq_d^i. \quad (2)$$

These values can also take on negative quantities to represent fixed generation.

Constant impedance and constant current loads are not implemented directly, but the constant impedance portions can be modeled as a shunt element described below. As mentioned above, dispatchable loads are modeled as negative generators and appear as negative values in s_g .

A shunt connected element such as a capacitor or inductor is modeled as a fixed impedance to ground at a bus. The admittance of the shunt element at bus i is given as

$$y_{sh}^i = g_{sh}^i + jb_{sh}^i.$$

Table B-1: Bus Data (`mpc.bus`)

name	column	description
BUS_I	1	bus number (positive integer)
BUS_TYPE	2	bus type (1 = PQ, 2 = PV, 3 = ref, 4 = isolated)
PD	3	real power demand (MW)
QD	4	reactive power demand (MVar)
GS	5	shunt conductance (MW demanded at $V = 1.0$ p.u.)
BS	6	shunt susceptance (MVar injected at $V = 1.0$ p.u.)
BUS_AREA	7	area number (positive integer)
VM	8	voltage magnitude (p.u.)
VA	9	voltage angle (degrees)
BASE_KV	10	base voltage (kV)
ZONE	11	loss zone (positive integer)
VMAX	12	maximum voltage magnitude (p.u.)
VMIN	13	minimum voltage magnitude (p.u.)
LAM_P [†]	14	Lagrange multiplier on real power mismatch (u /MW)
LAM_Q [†]	15	Lagrange multiplier on reactive power mismatch (u /MVar)
MU_VMAX [†]	16	Kuhn-Tucker multiplier on upper voltage limit (u /p.u.)
MU_VMIN [†]	17	Kuhn-Tucker multiplier on lower voltage limit (u /p.u.)

[†] Included in OPF output, typically not included (or ignored) in input matrix. Here we assume the objective function has units u .

Рис. 1 Bus data from Matlab MatPower

Данные о величинах p_g , q_g , p_d , q_d , g_{sh} , b_{sh} можно найти в таблице Рис.2, а соответствующие им поля расшифрованы в таблице Рис.1

```

%% bus data
%      bus_i  type  Pd   Qd   Gs   Bs   area  Vm   Va   baseKV  zone  Vmax  Vmin
mpc.bus = [
    1      3      0      0      0      0      1    1.06  0      0      1    1.06  0.94;
    2      2    21.7    12.7    0      0      1    1.045 -4.98  0      1    1.06  0.94;
    3      2    94.2     19      0      0      1    1.01  -12.72 0      1    1.06  0.94;
    4      1    47.8    -3.9      0      0      1    1.019 -10.33 0      1    1.06  0.94;
    5      1     7.6     1.6      0      0      1    1.02   -8.78  0      1    1.06  0.94;
    6      2    11.2     7.5      0      0      1    1.07  -14.22 0      1    1.06  0.94;
    7      1      0      0      0      0      1    1.062  -13.37 0      1    1.06  0.94;
    8      1      0      0      0      0      1    1.09  -13.36 0      1    1.06  0.94;
    9      1    29.5    16.6      0     19      1    1.056  -14.94 0      1    1.06  0.94;
   10      1     9      5.8      0      0      1    1.051  -15.1  0      1    1.06  0.94;
   11      1     3.5     1.8      0      0      1    1.057  -14.79 0      1    1.06  0.94;
   12      1     6.1     1.6      0      0      1    1.055  -15.07 0      1    1.06  0.94;
   13      1    13.5     5.8      0      0      1    1.05   -15.16 0      1    1.06  0.94;
   14      1    14.9      5      0      0      1    1.036  -16.04 0      1    1.06  0.94;
];

```

Рис. 2 Bus data case 14 from IEEE

Узлы соединены между собой линиями передач. All transmission lines, transformers and face shifters are modeled with a common branch model, consisting of standard Π transmission line model with series impedance $z_s = r_s + jx_s$ and total charging susceptance b_c , in series with an ideal phase shifting transformers.

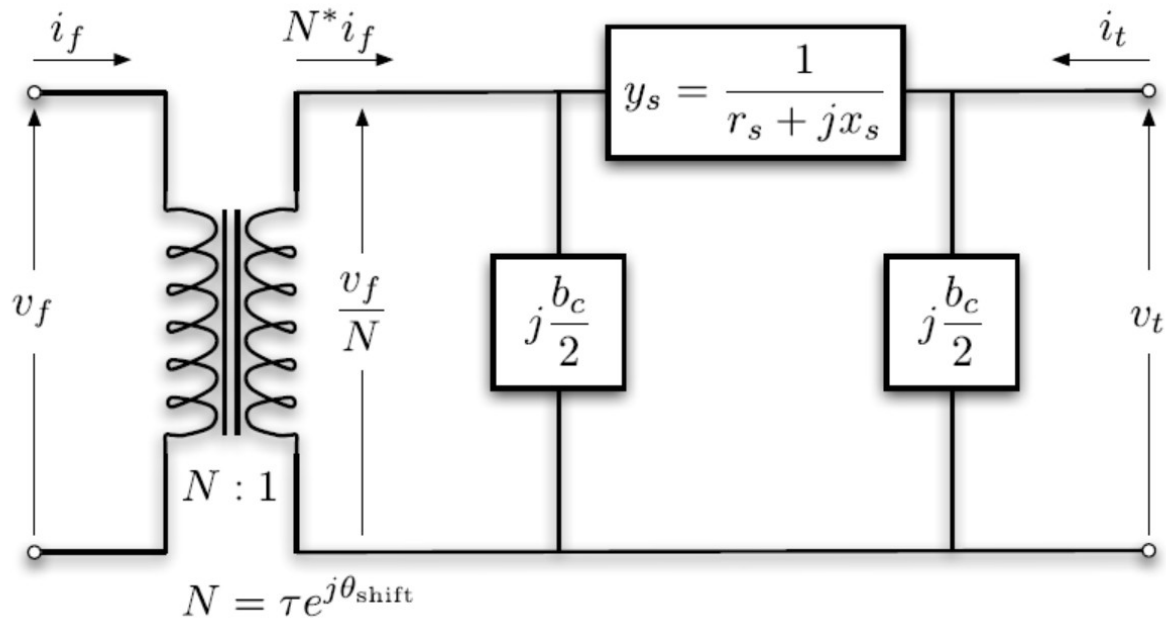


Рис. 3 Branch model

The complex power injection i_f and i_t at the from and to ends of the branch, respectively, can be expressed in terms of the 2×2 branch admittance matrix Y_{br} and the respective terminal voltages v_f and v_t

$$\begin{bmatrix} i_f \\ i_t \end{bmatrix} = Y_{br} \begin{bmatrix} v_f \\ v_t \end{bmatrix}, \quad \text{где} \quad Y_{br} = \begin{bmatrix} (y_s + j\frac{b_c}{2})\frac{1}{r^2} & -y_s\frac{1}{\tau e^{j\theta_{shift}}} \\ -y_s\frac{1}{\tau e^{j\theta_{shift}}} & y_s + j\frac{b_c}{2} \end{bmatrix}.$$

Данные о величинах для линий передач можно найти в таблице Рис.5, а расшифровка полей в таблице Рис.4.

name	column	description
F_BUS	1	“from” bus number
T_BUS	2	“to” bus number
BR_R	3	resistance (p.u.)
BR_X	4	reactance (p.u.)
BR_B	5	total line charging susceptance (p.u.)
RATE_A	6	MVA rating A (long term rating), set to 0 for unlimited
RATE_B	7	MVA rating B (short term rating), set to 0 for unlimited
RATE_C	8	MVA rating C (emergency rating), set to 0 for unlimited
TAP	9	transformer off nominal turns ratio, if non-zero (taps at “from” bus, impedance at “to” bus, i.e. if $r = x = b = 0$, $tap = \frac{ V_f }{ V_t }$; $tap = 0$ used to indicate transmission line rather than transformer, i.e. mathematically equivalent to transformer with $tap = 1$)
SHIFT	10	transformer phase shift angle (degrees), positive \Rightarrow delay
BR_STATUS	11	initial branch status, 1 = in-service, 0 = out-of-service
ANGMIN*	12	minimum angle difference, $\theta_f - \theta_t$ (degrees)
ANGMAX*	13	maximum angle difference, $\theta_f - \theta_t$ (degrees)
PF†	14	real power injected at “from” bus end (MW)
QF†	15	reactive power injected at “from” bus end (MVar)
PT†	16	real power injected at “to” bus end (MW)
QT†	17	reactive power injected at “to” bus end (MVar)
MU_SF‡	18	Kuhn-Tucker multiplier on MVA limit at “from” bus (u /MVA)
MU_ST‡	19	Kuhn-Tucker multiplier on MVA limit at “to” bus (u /MVA)
MU_ANGMIN‡	20	Kuhn-Tucker multiplier lower angle difference limit (u /degree)
MU_ANGMAX‡	21	Kuhn-Tucker multiplier upper angle difference limit (u /degree)

* Not included in version 1 case format. The voltage angle difference is taken to be unbounded below if $ANGMIN \leq -360$ and unbounded above if $ANGMAX \geq 360$. If both parameters are zero, the voltage angle difference is unconstrained.

† Included in power flow and OPF output, ignored on input.

‡ Included in OPF output, typically not included (or ignored) in input matrix. Here we assume the objective function has units u .

Рис. 4 Transmission line from Matlab MatPower


```

%% branch data
%      fbus  tbus   r      x      b      rateA  rateB  rateC  ratio  angle  status  angmin  angmax
mpc.branch = [
    1      2      0.01938 0.05917 0.0528 0        0        0        0        0        0        1       -360    360;
    1      5      0.05403 0.22304 0.0492 0        0        0        0        0        0        1       -360    360;
    2      3      0.04699 0.19797 0.0438 0        0        0        0        0        0        1       -360    360;
    2      4      0.05811 0.17632 0.034  0        0        0        0        0        0        1       -360    360;
    2      5      0.05695 0.17388 0.0346 0        0        0        0        0        0        1       -360    360;
    3      4      0.06701 0.17103 0.0128 0        0        0        0        0        0        1       -360    360;
    4      5      0.01335 0.04211 0        0        0        0        0        0        0        1       -360    360;
    4      7      0        0.20912 0        0        0        0        0.978 0        1       -360    360;
    4      9      0        0.55618 0        0        0        0        0.969 0        1       -360    360;
    5      6      0        0.25202 0        0        0        0        0.932 0        1       -360    360;
    6      11     0.09498 0.1989  0        0        0        0        0        0        1       -360    360;
    6      12     0.12291 0.25581 0        0        0        0        0        0        1       -360    360;
    6      13     0.06615 0.13027 0        0        0        0        0        0        1       -360    360;
    7      8      0        0.17615 0        0        0        0        0        0        1       -360    360;
    7      9      0        0.11001 0        0        0        0        0        0        1       -360    360;
    9      10     0.03181 0.0845  0        0        0        0        0        0        1       -360    360;
    9      14     0.12711 0.27038 0        0        0        0        0        0        1       -360    360;
    10     11     0.08205 0.19207 0        0        0        0        0        0        1       -360    360;
    12     13     0.22092 0.19988 0        0        0        0        0        0        1       -360    360;
    13     14     0.17093 0.34802 0        0        0        0        0        0        1       -360    360;
];

```

Рис. 5 Transmission line data Case 14 from IEEE

For a network with n buses, all constant impedance elements of the model are incorporated into a complex $n/timesn$ bus admittance matrix Y_{bus} that relates the complex nodal current injections I_{bus} to the complex node voltages V :

$$I_{bus} = Y_{bus}V.$$

Y_{bus} is an огромная sparse matrix, which учитывает все transmission lines и шунты, присутствующие в энергетической сети.

Вектор V содержит напряжения на всех узлах. Вектор I - соответствующие токи.

Тогда the current injection I_{bus} can be used to calculate the complex power injections as functions of the complex voltages V :

$$S_{bus}(V) = [V] I_{bus}^* = [V] Y_{bus}^* V^*.$$

The nodal bus injections are then matched to the injections from loads and generators to form the AC nodal power balance equations, expressed as a function of the complex bus voltage and generator injections in complex matrix form as

$$g_S(V, S_g) = S_{bus}(V) + S_d - C_g S_g, \quad (3)$$

где S_d это вектор, составленный из s_d (2), S_g это вектор, составленный из s_g (1), C_g - a sparse $n \times n$ generator connection matrix, where its (i, j)-th element is 1 if generator j is located as bus i and 0 otherwise. Привязка генераторов к узлам определяется таблицей¹ Рис.6.

```
%% generator data
```

%	bus	Pg	Qg	Qmax	Qmin	Vg	mBase	status	Pmax	Pmin
mpc.gen = [
	1	232.4	-16.9	10	0	1.06	100	1	332.4	0
	2	40	42.4	50	-40	1.045	100	1	140	0
	3	0	23.4	40	0	1.01	100	1	100	0
	6	0	12.2	24	-6	1.07	100	1	100	0
	8	0	17.4	24	-6	1.09	100	1	100	0
];										

Рис. 6 Generators data Case 14 from IEEE

3 Problem Statement

Consider a power network modeled by a directed graph $\mathbf{G}(\mathcal{N}, \mathcal{E})$ where buses are represented by nodes in \mathcal{N} and transmission lines by edges in $\mathcal{E} \subset \mathcal{N} \times \mathcal{N}$. Let \mathcal{N}_{slack} denote the set of slack buses², $\mathcal{N}_{ns} = \mathcal{N} \setminus \mathcal{N}_{slack}$ - the set of non-slack buses. The sets of PV³ and PQ⁴ buses are denoted by \mathcal{N}_{pv} and \mathcal{N}_{pq} respectively. Let $\mathcal{N}_G = \mathcal{N}_{pv} \cup \mathcal{N}_{slack}$ denote all generators.

Consider the following AC Power Flow equation (3) in polar form:

$$p_i^{inj} = \sum_{k \in \mathcal{N}} v_i v_k (G_{ik} \cos(\theta_{ik}) + B_{ik} \sin(\theta_{ik})) \quad (4)$$

$$q_i^{inj} = \sum_{k \in \mathcal{N}} v_i v_k (G_{ik} \sin(\theta_{ik}) - B_{ik} \cos(\theta_{ik})) \quad (5)$$

Consider the following operational constraints:

$$p_i^{min} \leq p_i^{inj} \leq p_i^{max}, \quad i \in \mathcal{N}_G \quad (6)$$

$$q_i^{min} \leq q_i^{inj} \leq q_i^{max}, \quad i \in \mathcal{N}_G \quad (7)$$

$$v_i^{min} \leq v_i^{inj} \leq v_i^{max}, \quad i \in \mathcal{N}_{pq} \quad (8)$$

$$\varphi_l^{min} \leq \theta_l^{from} - \theta_l^{to} \leq \varphi_l^{max}, \quad l \in \mathcal{E}, \quad (9)$$

where $\theta_l^{from} - \theta_l^{to}$ denotes the voltage phase difference between from and to ends of the transmission lines. The operational constraints considered here are power limits and voltage magnitude limits at the generators and phase angle difference limits on transmission lines.

¹Примеры данных, приведенные в таблицах Рис.2, Рис.5, Рис.6, взяты из файла <https://github.com/lanl-ansi/PowerModels.jl/blob/master/test/data/matpower/case14.m>

²Что такое slack

³про PV

⁴про PQ

In analysis of power grid the system operator has control over the generator powers, which are denoted by $u = p_{ns}^{inj}$. The corresponding internal states are $x = [\theta_{ns}^T v_{pq}^T]^T$. The system operator needs to decide the set of control variables u , subject to the power flow feasibility set in equations (4)-(9). Чтобы проверить условие feasibility необходимо определить state variables x по заданным control variables u — то есть решить задачу general AC power flow problem. Так как уравнения (4) и (5) нелинейные по компонентам вектора x .

В литературе описаны несколько итерационных методов для решения такой задачи. Итерация осуществляется по формуле

$$x = -J_{f,0}^{-1} M g(x, u), \quad (10)$$

где $J_{f,0}$ - якобиан рассчитывается в начальной точке x_0, y_0 , для которой выполняются все условия feasibility (4)-(9). Уравнение (10) записано для общей системы нелинейных равенств $f(x, u) = 0$, где $u \in \mathbb{R}^m$, $x \in \mathbb{R}^n$. Введем базисные функции $\psi : (\mathbb{R}^n, \mathbb{R}^m) \rightarrow \mathbb{R}^q$ так, что

$$f(x, u) = M \psi(x, u) \quad (11)$$

где $M \in \mathbb{R}^{n \times q}$ — постоянная матрица. Базовыми функциями удобно пользоваться, так как каждая базовая функция ψ_k зависит от малого числа компонент вектора x . Введем новую функцию $g(x, u)$ as follows:

$$g(x, u) = \psi(x, u) - J_{\psi,0} x, \quad \text{where } J_{\psi,0} = \left. \frac{\partial \psi}{\partial x} \right|_{x_0, u_0}. \quad (12)$$

Note that $J_{f,0} = M J_{\psi,0}$. That приводит к тому, что уравнение (10) эквивалентно уравнениям (4)-(5) при следующем выборе вектора функций ψ :

$$\psi(x, u) = \begin{bmatrix} p_i^{inj} \\ q_i^{inj} \\ v^f v^t \cos(\theta^f - \theta^t) \\ v^f v^t \sin(\theta^f - \theta^t) \\ (v^f)^2 \end{bmatrix} \quad (13)$$

Оператор изменяет control variables не только с целью обеспечить выполнение неравенств (6)-(9), но и обеспечить минимальную стоимость производства энергии. Для этого необходимо решить AC - Optimal Power Flow Problem, например минимизировать функционал:

$$\sum_{k \in \mathcal{N}_G} (c_{2k}((p_k^g)^2 + c_{1k}(p_k^g) + c_{0k}) \longrightarrow \min_{p^g} \quad (14)$$

where c_{2k}, c_{1k}, c_{0k} , $\forall k \in \mathcal{N}_G$ — generator cost components. Функционал может учитывать и стоимость реактивной мощности и другие параметры, а также быть записан в форме кусочно-непрерывной функции.

4 Implementation overview

Численное решение задачи AC-OPF можно разбить на следующие этапы. Каждый этап подразумевает решение отдельной задачи. В целом, эти задачи могут решаться несколько раз на разных этапах и итерациях общего алгоритма. Некоторые задачи, например 3, 5, 6 и 7, представляют самостоятельный научный интерес и уже реализованы в ряде программных пакетов.

1. Задача адаптации и верификации исходных данных. Включает проверку данных из различных источников с различными форматами, проверку их целостность, составление сводки удобного формата для дальнейшей работы. Необходимость этого обусловлена тем, что алгоритм имеет особенности, такие как разделение вершин по типам (PQ, PV, slack) и т.д. Также подразумевает задание функционала для оптимизации. Его тоже нужно верифицировать, так как он должен быть только в переменных u и использовать все переменные u .
2. Задача определения начальной точки. В тривиальном случае она задается оператором. Точка должна быть проверена по плану, описанному в 7.
3. Задача построения выпуклого сужения согласно статье [??].
4. Задача анализа области сужения. Включает отрисовку для исследовательских целей двумерных срезов невыпуклой области и выпуклого сужения в независимых линейных комбинациях переменных u . Также допускает отрисовку последовательности точек, полученных итерационным алгоритмом.
5. Оптимизация функционала по выпуклой области. Решается в переменных u . Методов поиска оптимума может быть несколько.
6. Задача восстановления всех переменных, описывающих систему, по control variables (u). В статье [??] приводится алгоритм, описывающий итерации метода Ньютона-Рафсона.
7. Задача построения лучшей проекции на условия задачи. Это необходимо, если точка, полученная в 6 оказалась за границами feasibility set.
8. Необходимо создать программный продукт, осуществляющий managing программ, решающих перечисленные выше задачи. Trust region approach. Выполняет по очереди: 1, 2, 7, 3, 5, 6, 7, потом проверку качества решения, и, если оно недостаточно качественное, то возвращается к 3.

Особый интерес представляет задача восстановления state variables (x) по заданным control variables (u). Действительно, располагая алгоритмом решения этой задачи и достаточным вычислительным ресурсом, любая задача связанная с Power Flow может быть решена простым перебором по сетке. Понятно, что так не поступают из-за огромной размерности задачи. Однако в ряде случаев решение перебором может быть полезно, например, для построения сечения feasible области. Это важно при исследовании новых алгоритмов для визуального сравнения результатов и валидации окончательного результата работы полного алгоритма решения задачи AC-OPF.

5 Экономный алгоритм выделения границы feasible set

Для простоты формулировки алгоритма рассмотрим двумерный случай.

Пусть заданы две точки A и B, первая внутри feasible set, вторая снаружи.

TODO: описание алгоритма

TODO: описание построения какой-то точки снаружи B по точке внутри области A.

Описанный алгоритм легко обобщается на многомерный случай.

Рассмотрим результаты работы алгоритма восстановления границы на примере невыпуклой области, изображенной на Рис.7 в левом верхнем углу. Эта область в контексте данной работы моделирует feasible set. Эта область получена в результате пересечения областей, ограниченных тремя гиперболами. Стартовая внутренняя точка имеет координаты (5, 5). А стартовая внешняя — (5, 15). Таким образом, на первой итерации длина сегмента составляет 10 единиц. На каждой следующей итерации длина сегмента сокращается вдвое

по сравнению с предыдущим. На рисунке Рис.8 показаны восстановленные границы для старших итераций. Видно, что каждый сегмент пересекает границу области.

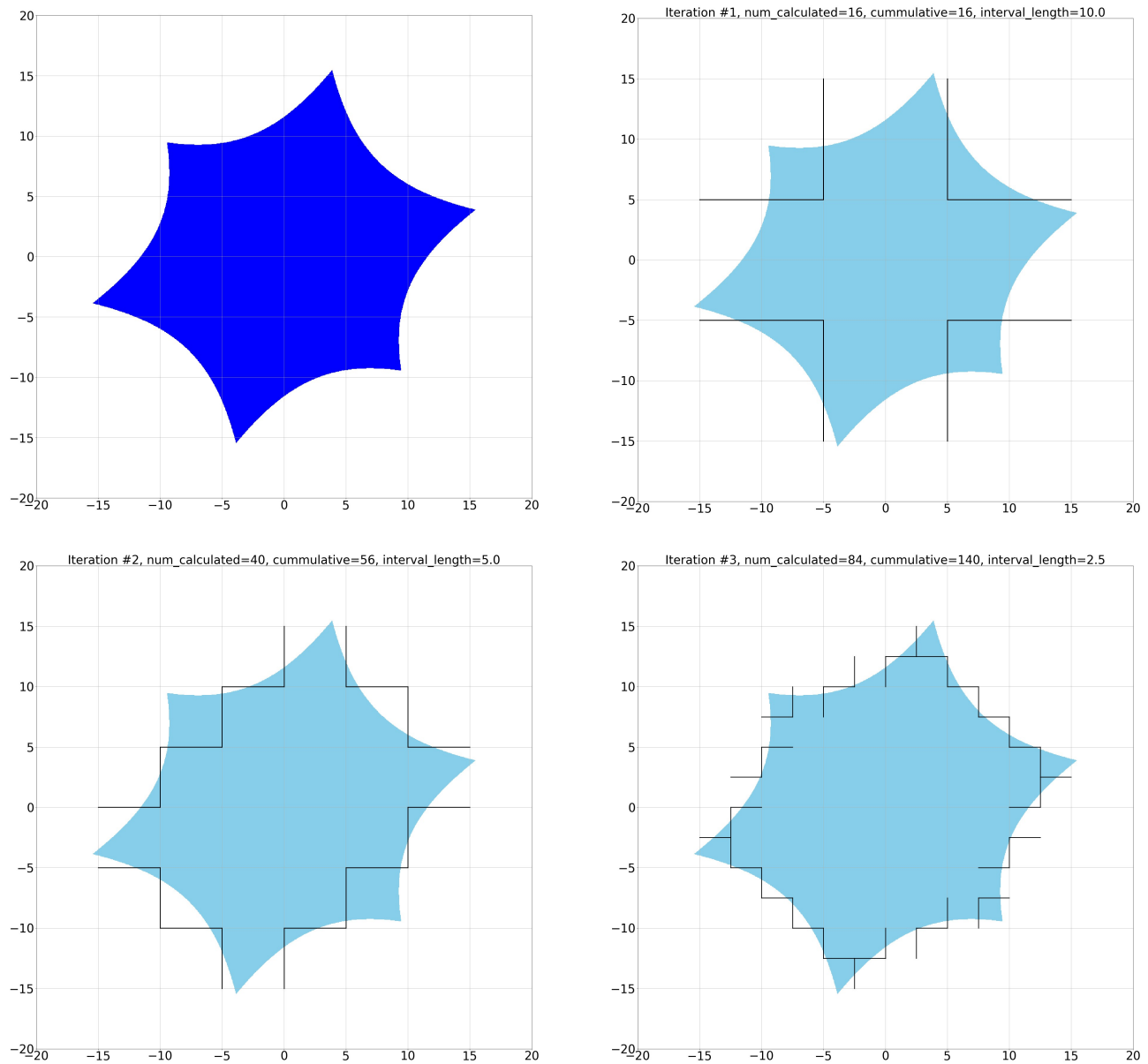


Рис. 7 Первые итерации алгоритма восстановления границы. Синяя область — область, задаваемая ограничениями. Сегменты показаны черными отрезками. Над графиками указаны номер итерации, количество точек, обчисленных на данной итерации, сумма количества обчисленных точек с учетом предыдущих итераций. Координаты области указаны в условных единицах.

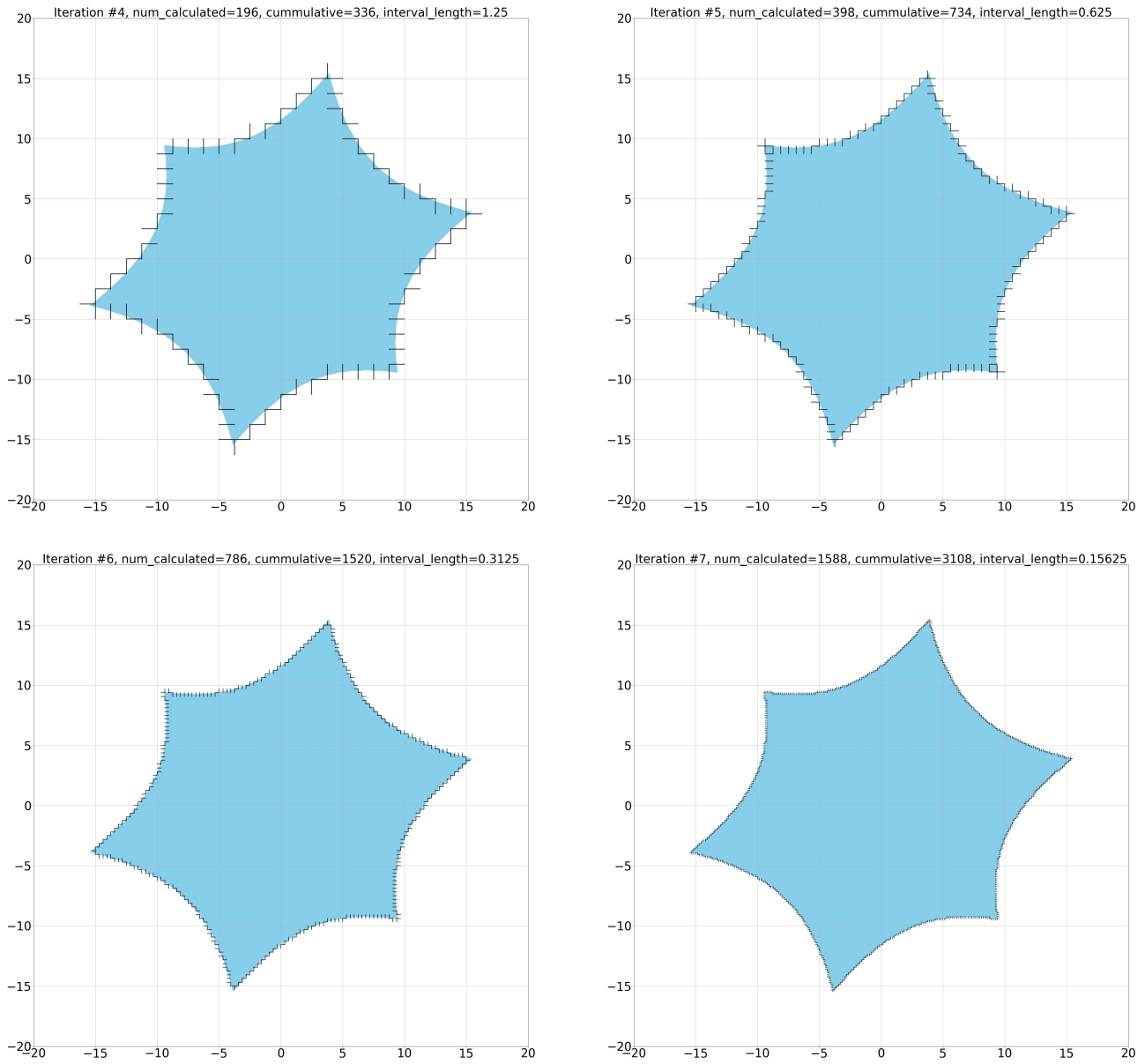


Рис. 8 Старшие итерации алгоритма восстановления границы. Размер сегмента с каждой итерацией уменьшается вдвое. На 8 итерации размер сегмента составляет 0.15625 условных единиц. Видно, что облако сегментов визуально не отличается от границы области.

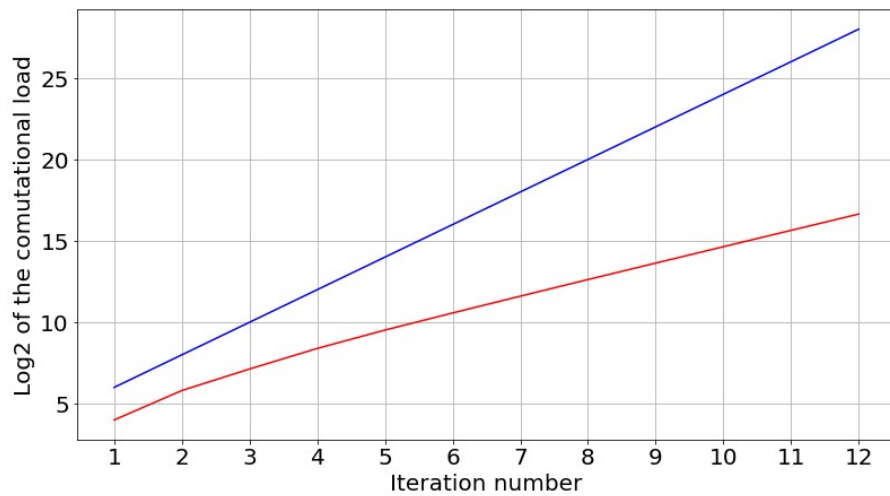


Рис. 9 График сравнения эффективности экономного алгоритма (красная линия) и алгоритма перебора по сетке (синяя линия). Отношение эффективности алгоритмов на каждой итерации составляет 2^{Δ} , где Δ — разность между графиками.

Рассмотрим эффективность данного алгоритма. Для этого сравним вычислительную сложность предложенного алгоритма и алгоритма перебора по сетке при одинаковых длине сегмента шага сетки. Вычислительная сложность измеряется в количестве точек, для которых проверяется ее принадлежность *feasible set*. Так как вычислительный ресурс, который приходится тратить на эту задачу, существенно преобладает над другими операциями. Напомним, что задача проверки принадлежности точки *feasible set* включает в себя итерационную процедуру восстановления *state variables* по *control variables*.

Вычислительная сложность расчетов по сетке обратно пропорциональна квадрату шага сетки. Поэтому, если на каждой итерации мы уменьшаем шаг вдвое, то вычислительная сложность увеличивается в 4 раза. Для рассмотренного примера график логарифма вычислительной сложности в зависимости от номера итерации показан на Рис.15 синей линией. Аналогичный график для экономного алгоритма показан красной линией. Относительная эффективность алгоритма на каждой итерации отражается разностью между двумя графиками. Так, на 4-й итерации при длине шага 1.25 эффективность экономного алгоритма в $2^{3.607} \simeq 12.2$ раза превышает эффективность простого перебора по сетке. Для 8-й итерации эффективность будет 166.3 раза, а для 12-й итерации — 2619.9 раз.

6 Использование алгоритма поиска границы для решения AC-OPF

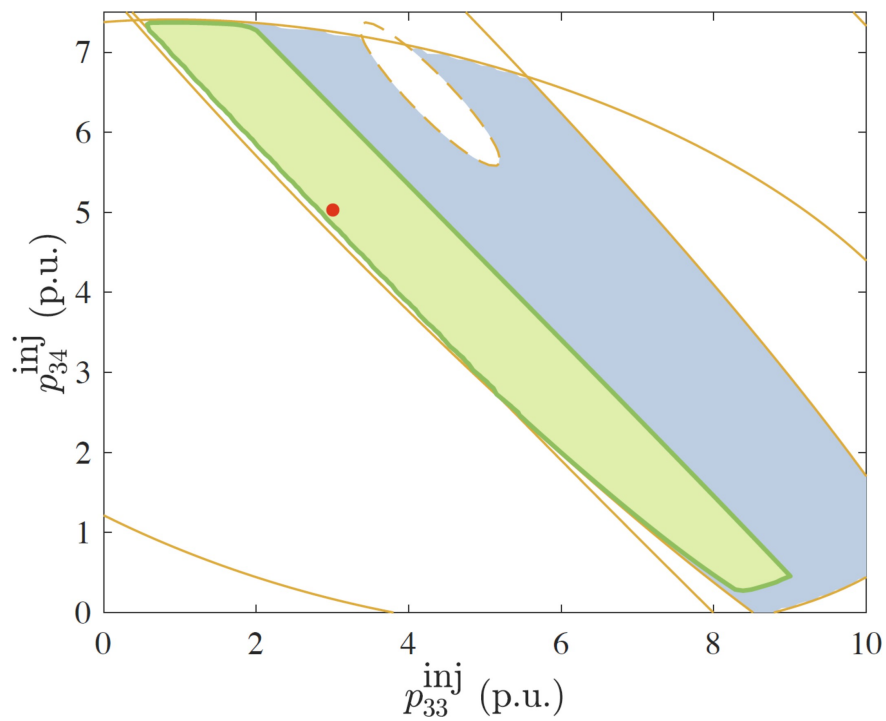


Рис. 10 Пример среза feasible set из IEEE 39 bus

This section provides the visualization and analysis of approximate solution of AC-OPF Problem for IEEE 39 bus system (Рис.16) using the алгоритм восстановления control variables по state variables and the algorithm introduced at the previous chapter. В 2d срезе по control variables p_{33}^{inj} and p_{34}^{inj} the blue and green area represents feasibility set, orange lines - границы областей, задающих отдельные ограничения. The results are shown on Рис.11 and Рис.12.

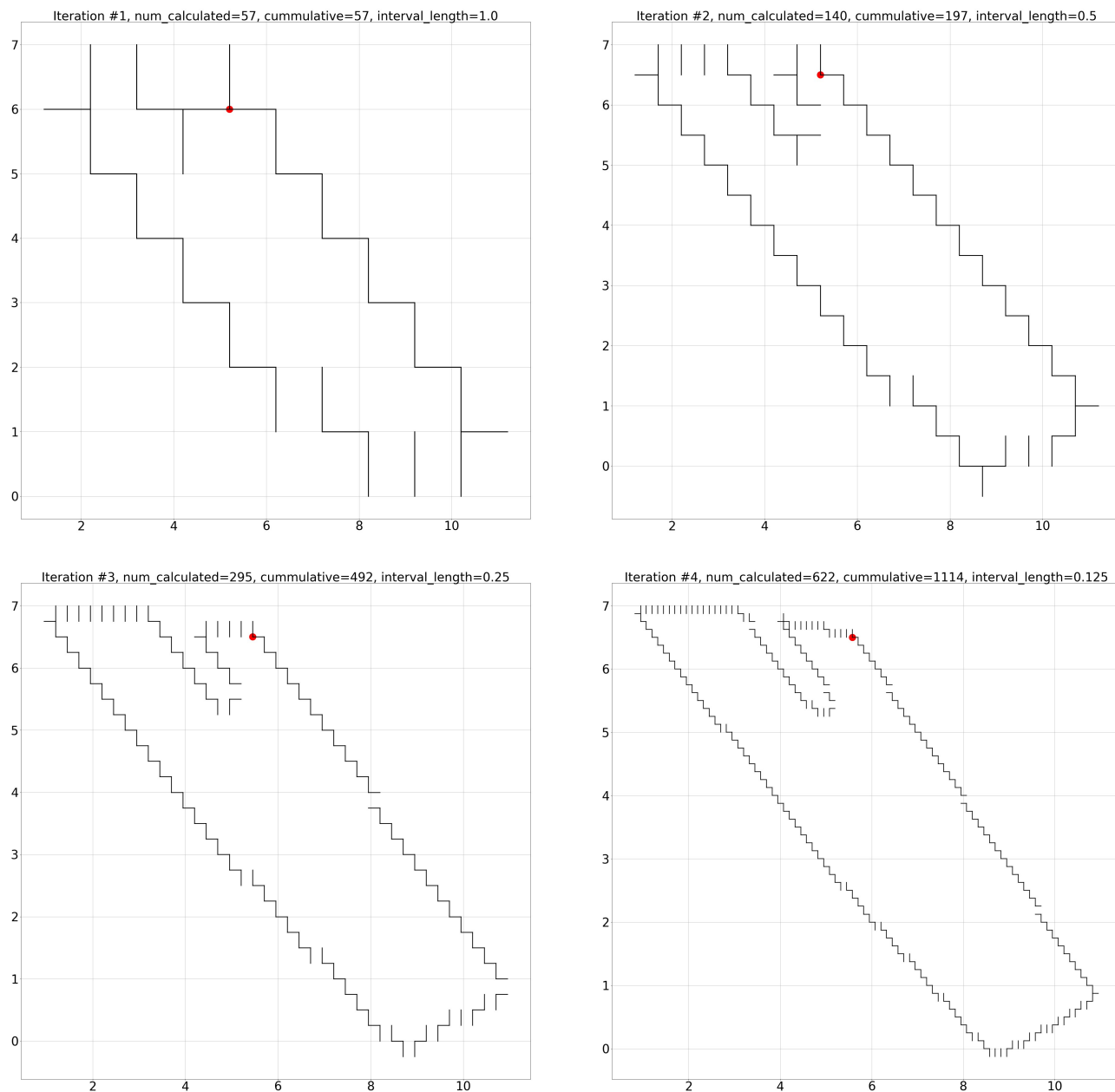


Рис. 11 Результат первых итераций восстановления границы feasible set (черные сегменты) и точки (красный цвет), соответствующей минимуму функционала на внутренних граничных точках выделенных сегментов.

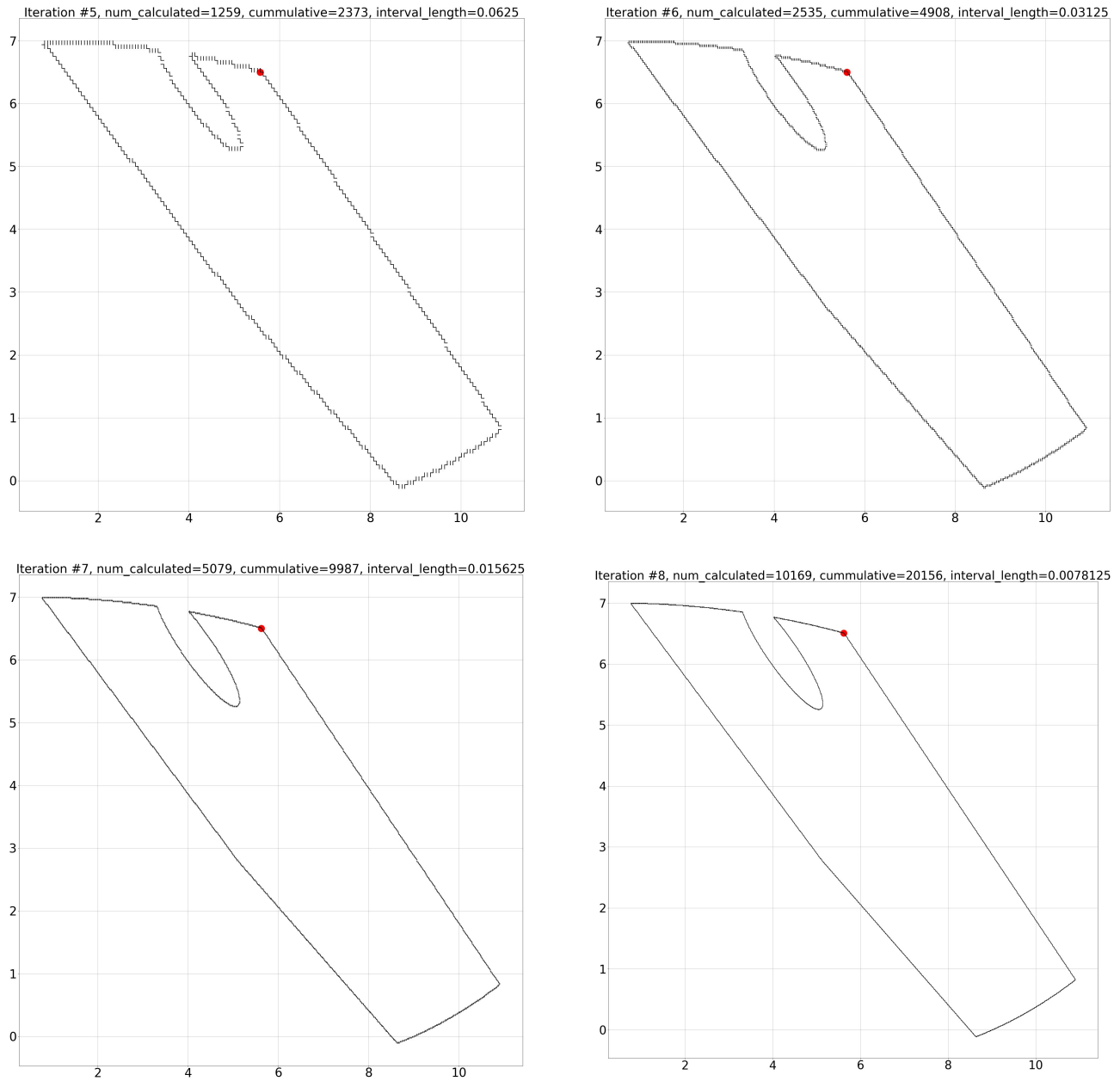


Рис. 12 Результаты старших итераций для IEEE 39. Видно, что множество сегментов визуально образуем контур границы, совпадающий с границей на Рис.16.

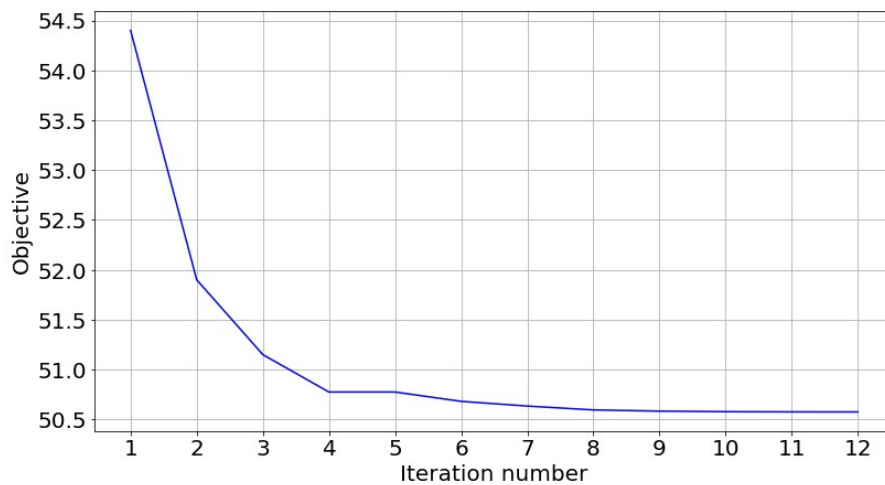


Рис. 13 По оси абсцисс отложен номер итерации. По оси ординат — значение функционала в лучшей внутренней точке множества границ сегментов.

7 Conclusion

В работе рассмотрена задача power flow. Эта задача имеет важное практическое значение для оптимизации работы энергосетей. Основная цель оптимизации состоит в удешевлении выработки электроэнергии. Диспетчер может управлять мощностью генераторов, которые связаны между собой и потребителями линиями передач. Каждый генератор, потребитель и линия передач имеют собственные характеристики. Некоторые параметры определяются техническими характеристиками например сопротивление и емкость линий передач, а некоторые эксплуатационными, например, минимальные и максимальные значения вырабатываемой мощности. Все эти данные сведены в таблицы, согласно стандарту IEEE??. В обзорной главе номер 2 описана структура и назначение этих данных. В этой же главе описано построение графа энергетической сети. Математическая постановка задачи включает в себя набор ограничений и целевую функцию. Часть ограничений - уравнения содержащие строгие равенства, а другая часть ограничений - неравенства. Сложность задачи обусловлена двумя вещами. Во-первых, эти уравнения ограничений нелинейные, во-вторых, что задача имеет большую размерность. Существует несколько подходов к решению этой задачи. Один из них основан на выделении внутренней выпуклой области внутри feasible set, содержащей текущую рабочую точку. В границах этой области решается оптимизационная задача минимизации функционала. Полученная точка используется как стартовая для следующей итерации. Повторение этой процедуры, согласно последовательности, описанной в главе 4 приводит к решению задачи AC-OPF. Эта задача может быть разбита на несколько отдельных задач. Одна из них - восстановление state variable по набору control variables. При этом нужно обеспечить, что набор полученных значений удовлетворяет всем ограничениям задачи. В работе предложен экономный способ построения границы невыпуклой области, основанный на процедуре проверки существования state variables для заданных control variables, таких что точка, задаваемая обоими множествами этих переменных, принадлежит feasibility set. Для начала работы алгоритма достаточно указать две точки - одна внутри feasible области, другая снаружи. С каждой следующей итерацией длина сегмента, пересекающего границу, уменьшается вдвое. Процедура формирования сегмента обеспечивает, что каждый из них отстоит друг от друга не более чем на длину этого сегмента и плотно покрывают

границу области. Отметим, что для восстановления границы не требуется явно выделять значимые ограничения, а также не требуется указывать область, все множество feasibility. Так как каждый сегмент содержит гранитную точку принадлежащую множеству feasibility, мы можем решить оптимизационную задачу на множестве этих точек. В работе рассмотрен пример из case39, иллюстрирующий работу алгоритма. Несмотря на существенную экономию расчетного ресурса по сравнению с простым перебором по сетке, однако для решения задачи большой размерности вычислительные затраты чрезвычайно велики. Метод предназначен для использования в исследовательских целях, чтобы посмотреть насколько хорошо работает алгоритм построения сужения или для валидации результата в окрестно полученной точки результата.

8 *

Список литературы

- [1] Ian A. Hiskens Daniel K. Molzahn. A survey of relaxations and approximations of the power flow equations. foundations and trends. *Electric Energy Systems*, 4(1–2):1–221, 2019.
- [2] Steven H Low. Convex relaxation of optimal power flow—part i: Formulations and equivalence. *IEEE Transactions on Control of Network Systems*, 1(1):15–27, 2014.
- [3] Javad Lavaei and Steven H Low. Zero duality gap in optimal power flow problem. *IEEE Transactions on Power Systems*, 27(1):92–107, 2012.
- [4] Carleton Coffrin, Hassan L Hijazi, and Pascal Van Hentenryck. The qc relaxation: A theoretical and computational study on optimal power flow. *IEEE Transactions on Power Systems*, 31(4):3008–3018, 2016.
- [5] Steven H Low. Convex relaxation of optimal power flow—part ii: Exactness. *IEEE Transactions on Control of Network Systems*, 1(2):177–189, 2014.