ISSN 0005-1179 (print), ISSN 1608-3032 (online), Automation and Remote Control, 2024, Vol. 85, No. 6, pp. 598–609.
© The Author(s), 2024 published by Trapeznikov Institute of Control Sciences, Russian Academy of Sciences, 2024.
Russian Text © The Author(s), 2024, published in Avtomatika i Telemekhanika, 2024, No. 6, pp. 67–82.

TOPICAL ISSUE

An Algorithm for Finding the Generalized Chebyshev Center of Sets Defined via Their Support Functions

P. A. Arkhipov

Institute of Science and Technology Austria
e-mail: pavel.arkhipov@ist.ac.at

Received January 25, 2024
Revised March 12, 2024
Accepted March 20, 2024

Abstract—This paper is dedicated to an optimization problem. Let $A, B \subset \mathbb{R}^n$ be compact convex sets. Consider the minimal number $t^0 > 0$ such that $t^0 B$ covers A after a shift to a vector $x^0 \in \mathbb{R}^n$. The goal is to find t^0 and x^0. In the special case of B being a unit ball centered at zero, x^0 and t^0 are known as the Chebyshev center and the Chebyshev radius of A. This paper focuses on the case in which A and B are defined with their black-box support functions. An algorithm for solving such problems efficiently is suggested. The algorithm has a superlinear convergence rate, and it can solve hundred-dimensional test problems in a reasonable time, but some additional conditions on A and B are required to guarantee the presence of convergence. Additionally, the behavior of the algorithm for a simple special case is investigated, which leads to a number of theoretical results. Perturbations of this special case are also studied.

Keywords: optimization, Chebyshev center, gradient descent

DOI: 10.31857/S0005117924060053

1. INTRODUCTION

Let A and B be compact convex sets in $\mathbb{R}^n$. Let $t^0 \in \mathbb{R}$ be such a minimal positive number that some shifted copy of $t^0 B$ covers A,

$$t^0 = \min\{t \mid \exists x^0 \in \mathbb{R}^n : \ x^0 + t^0 B \supset A\}. \tag{1}$$

Figure 1 shows an example of the optimal configuration of A and B.

One may notice that the same problem could be stated if A is not convex. Indeed, if in such a problem A is replaced with its convex hull $conv(A)$, the optimal x^0 and t^0 remain the same. So, we can assume that A is convex without loss of generality.

The problem of finding the Chebyshev center of a set is a special case of the problem described above: if $B = B_1(0) = \{y \in \mathbb{R}^n \mid \|y\| \leqslant 1\}$, then x^0 is the Chebyshev center of A. It is known that it can be solved fast if A is finite [4]. But in general, Chebyshev-center-like problems turn out to be computationally challenging tasks, which received some attention in recent years. For example, finding the Chebyshev center of an intersection of balls is NP-hard [5] (2021). Even calculating the Chebyshev center of an intersection of two ellipsoids is a substantive problem [16] (2020). Chebyshev center draws such attention for a reason: it has a number of real-life applications. It can be helpful in Long-Term Hydrothermal Scheduling problems [17] (2022), regression problems with noisy terms [18] (2007), identification of linear dynamic systems with noisy parts [19] (2012), cybersecurity [20] (2023), [22] (2020) and robotics [21] (2021).

Let us consider such problems that the optimal x^0 is unique.

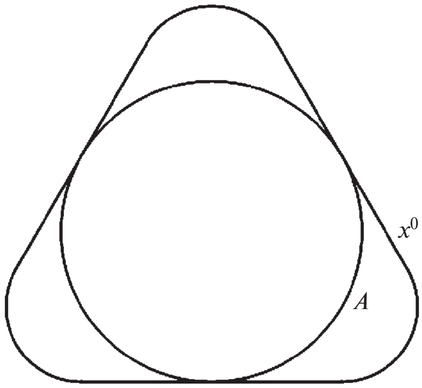

Fig. 1. The optimal configuration of A and B.

The goal is to develop an algorithm that solves the problem efficiently, if the sets A and B are defined with their support functions. Let us remind the reader that for a given set $M \subset \mathbb{R}^n$ the value of the support function of M at the point $p \in \mathbb{R}^n$ is defined as follows:

$$h_M(p) = \sup\{(p, y) \mid y \in M\}.$$

In this notation,

$$A = \{y \in \mathbb{R}^n \mid (y, p) \leqslant h_A(p)\ \forall p \in S^{n-1}\}, \tag{2}$$

$$B = \{y \in \mathbb{R}^n \mid (y, p) \leqslant h_B(p)\ \forall p \in S^{n-1}\}, \tag{3}$$

where $S^{n-1} = \{y \in \mathbb{R}^n \mid \|y\| = 1\}$ is the standard unit sphere. Then, the problem (1) can be restated:

$$t^0 = \min\{t \mid \exists x^0 \in \mathbb{R}^n :\ (x, p) + t \cdot h_B(p) \geqslant h_A(p)\ \forall p \in S^{n-1}\}. \tag{4}$$

In some sense, it is a linear programming problem with an infinite number of restrictions. Indeed, one minimizes a linear functional $(w, c) = t$, $w = (t, x_1, \dots x_n)^T \in \mathbb{R}^{n+1}$, $c = (1, 0, \dots, 0)^T$, subject to constraints $(w, b_p) \leqslant h_A(p)$, where $b_p = (h_B(p), p_1, \dots, p_n)^T$, that must hold for every p of unit norm.

One known approach to the problem is described in [1]. It picks a finite number of linear restrictions corresponding to a grid on the unit sphere and solves a linear programming problem subject to those restrictions. The disadvantage of this approach is that a reasonably fine grid on a high-dimensional unit sphere has too many elements. In practice, the computations become unbearably hard for $n > 4$. So, developing an algorithm that can solve the stated problem in high dimensions is a relevant research problem.

Before describing the algorithm, it makes sense to remind the reader that for a strictly convex compact set $M \subset \mathbb{R}^n$ its support function has the following gradient:

$$\nabla h_M(p) = \arg\max_{y \in M}(p, y). \tag{5}$$

Obviously, $(p, y) = h_M(p)$. If M is convex, but not strictly convex, one can consider the subdifferential $\partial h_M(p)$:

$$\partial h_M(p) = H(p) \cap M, \tag{6}$$

where $H(p) = \{y \in \mathbb{R}^n | (p, y) = h_M(p)\}$. The subdifferential of a convex function $f : \mathbb{R}^n \to \mathbb{R}$ at x_0 is the set of all vectors $v \in \mathbb{R}^n$ such that $f(x) - f(x_0) \geqslant (v, x - x_0)$ for all $x \in \mathbb{R}^n$. Such vectors v are called subgradients of f at x_0. Easy to see that if M is strictly convex, then the subdifferential of its support function is a one-element set at every point:

$$\partial h_M(p) = H(p) \cap M = \{y \in M | (p, y) = h_M(p)\} = \left\{\arg\max_{y \in M}(p, y)\right\} = \{\nabla h_M(p)\}. \tag{7}$$

Let us denote the convex hull of a set M as $conv(M)$, and the r-strongly convex hull of M as $strconv_r(M)$. A set is called r-strongly convex if it can be represented as an intersection of balls of radius r. Suppose $M \subset \mathbb{R}^n$ can be enclosed in a ball of radius r. Then, $strconv_r(M)$ is the intersection of all the balls of radius r that contain M (just like the convex hull of M is the intersection of all the half spaces that contain M).

2. THE ALGORITHM AND THEORETICAL RESULTS

For the algorithm described below to converge, A and B will have to satisfy a condition: the function $f : S^{n-1} \to \mathbb{R}$, $f(p) = (x^0, p) + t^0 h_B(p) - h_A(p)$ must have exactly $n+1$ local minimums $p_1^0, \ldots, p_{n+1}^0$, and the convex hull of the set of those minimums have to contain zero in its interior: $0 \in int(conv(p_1^0, \ldots, p_{n+1}^0))$. Informally, the conditions tell us that the linear programming problem (4) has exactly $n+1$ active restrictions

Let $p_1, \ldots, p_{n+1}$ be vectors of unit length. We will use the following notation:

$$a_i = h_A(p_i), \qquad a = (a_1, \ldots, a_{n+1})^T, \tag{8}$$

$$b_i = h_B(p_i), \qquad b = (b_1, \ldots, b_{n+1})^T, \tag{9}$$

$$M_a = \begin{pmatrix} a_1 & (p_1)_1 & \ldots & (p_1)_n \\ a_2 & (p_2)_1 & \ldots & (p_2)_n \\ \vdots & \vdots & \ddots & \vdots \\ a_{n+1} & (p_{n+1})_1 & \ldots & (p_{n+1})_n \end{pmatrix}, \tag{10}$$

$$M_b = \begin{pmatrix} b_1 & (p_1)_1 & \ldots & (p_1)_n \\ b_2 & (p_2)_1 & \ldots & (p_2)_n \\ \vdots & \vdots & \ddots & \vdots \\ b_{n+1} & (p_{n+1})_1 & \ldots & (p_{n+1})_n \end{pmatrix}. \tag{11}$$

The following lemma is obvious:

Lemma 1. *If $0 \in conv(\{p_1, \ldots, p_{n+1}\})$, then for t, x such that $(p_j, x) + t h_B(p_j) \geqslant h_A(p_j)$, and t is minimal, the following equality holds:*

$$M_b \begin{pmatrix} t \\ x_1 \\ \vdots \\ x_n \end{pmatrix} = a. \tag{12}$$

Corollary 1. *Using Cramer's rule,*

$$t = \frac{\det M_a}{\det M_b}. \tag{13}$$

Algorithm 1. Start with points $p_1, \ldots, p_{n+1} \in S^{n-1}$ equal to the vertices of a randomly rotated regular simplex. At each iteration, do the following:

1) Find the solution of the system of linear equations (12), t, x.

2) Substitute each p_i with p'_i, which is a local minimum of the function $f_{t,x}(p) = (x,p)+th_B(p)-h_A(p)$ found with a gradient descent starting from p_i. The gradient of $f_{t,x}$ can be computed as follows:

$$\nabla f_{t,x}(p) = x + t \arg\max_{y\in B}(p,y) - \arg\max_{y\in A}(p,y). \tag{14}$$

In case there are several global maximums of (p,y), choose an arbitrary one.

3) If there are no duplicates among p'_i, set p_i equal to p'_i and finish this iteration. Otherwise, remove duplicates from $\{p'_i\}$. Let L be the set of the remaining minimums p'_i. Perform a gradient descent to a minimum of $f_{t,x}$ starting from a (uniformly) random point of unit norm. If this minimum does not coincide with any element of L, add it to L. Do this until L has $n+1$ elements. Then, set p_i equal to p'_i and finish this iteration. If the number of performed tries is greater than K, but we still have not found $n+1$ distinct minimums, restart the algorithm.

Stop when t and x change little enough over one iteration, or when the number of iterations exceeds some limit.

Next, let us discuss the types of problems that can be solved with Algorithm 1.

Lemma 2. *If B is strictly convex, then the function $f_{t,x} = (x,p) + th_B(p) - h_A(p)$, $t \in \mathbb{R}^+$, $x \in \mathbb{R}^n$ is differentiable at every minimum.*

In particular, $f_{t,x}$ is differentiable at its minimums for the problems of finding the Chebyshev center. The differentiability of $f_{t,x}$ at its minimums may also be helpful for the gradient descent to converge properly. But, there are modifications of the gradient descent approach that can converge to non-smooth minimums.

Theorem 1. *Let $p_1^0, \dots, p_{n+1}^0$ be the $n+1$ minimums of $f(p) = (x^0,p) + t^0 h_B(p) - h_A(p)$, $0 \in int(conv(p_1^0,\dots,p_{n+1}^0))$. Let B be strictly convex. Suppose that for any $j \in \{1\dots n+1\}$ for any point p from some neighborhood of p_j^0,*

$$M\|p - p_j^0\|^\alpha \geqslant f(p) \geqslant \mu\|p - p_j^0\|^2, \tag{15}$$

where M, μ, α are constants, $1 < \alpha \leqslant 2$. Then, in some neighborhood of the solution, Algorithm 1 converges with order α. So, for t, x from some neighborhood of t^0, x^0,

$$\|(t'-t^0, x'_1 - x_1^0, \dots, x'_n - x_n^0)^T\| \leqslant C\|(t-t^0, x_1 - x_1^0, \dots, x_n - x_n^0)^T\|^\alpha, \tag{16}$$

where t', x' is the state of the algorithm at the next iteration, if the current state is t, x.

Theorem 2. *Let $B = B_1(0)$, $A = conv((v_1 + C_1)\cup\dots\cup(v_{n+1}+C_{n+1}))$, where $v_j \in \mathbb{R}^n$, $C_j \subset \mathbb{R}^n$ are a strongly convex sets with the radius of strong convexity $r < t^0$. Also, let f have at least $n+1$ zeroes. Then Algorithm 1 has quadratic convergence.*

3. NUMERICAL EXPERIMENTS

The code that was used for the numerical experiments, as well as the figures, are publically available the GitHub page, https://github.com/Paul566/GraduateThesishttps://github.com/Paul566/GraduateThesis. The solver class is in the file 'GradientDescentSolver.py'.

Algorithm 1 was implemented and tested. During all the tests, 15 iterations were enough for the algorithm to converge to machine precision. In all cases, the error of the final result was around 10^{-14}–10^{-16}. In most cases, 5–7 iterations were sufficient.

In all tests, the algorithm had the following parameters: maximal number of iterations: 20, gradient descent learning rate: 1, the number of attempts to find $n+1$ minimums: $10n$, maximal

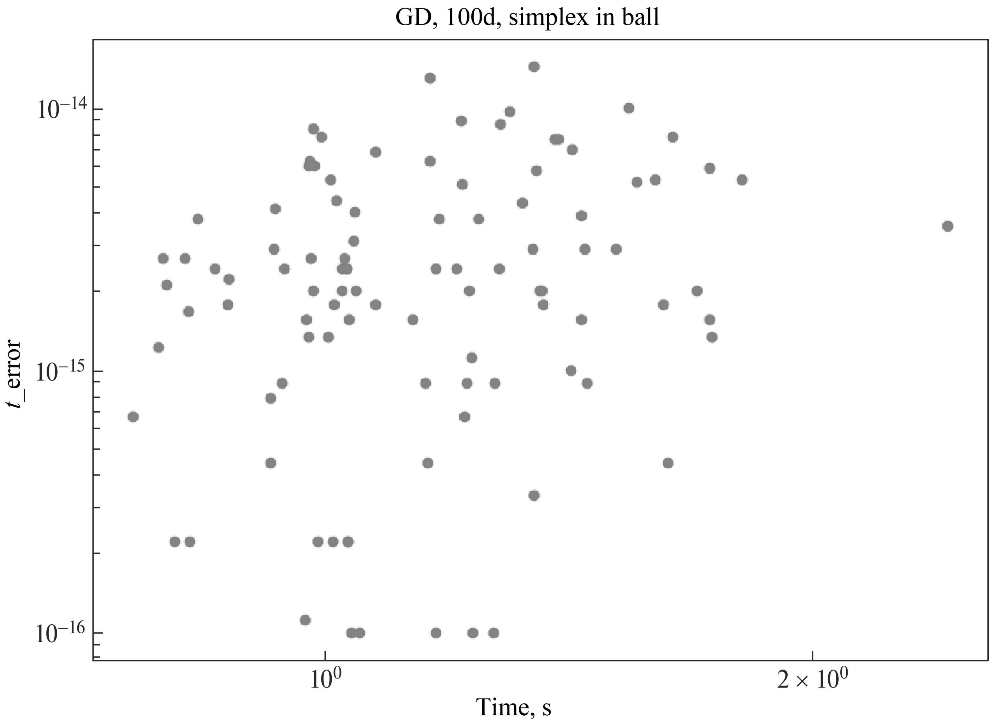

Fig. 2. Error of t versus runtime, $n = 100$, results of 100 tests.

number of steps during gradient descent: 1000, the gradient descent terminates if f changes by less than 10^{-10}, two minimums are considered to be duplicates if the norm of their difference is less than 10^{-6}, maximal number of restarts: 1000. The standard gradient descent without any modifications was used. For some test series, a plot of the error of the output versus runtime will be presented. For the simulations, an ordinary laptop without GPU acceleration was used.

1) Let us begin with a very simple series of tests. Let $B = B_1(0)$, A be a random simplex that contains the center of its circumscribed sphere. The results of the numerical experiments for dimension 100 is presented in Fig. 2. Of course, solving such problems with Algorithm 1 makes no practical sense, this series of simulations was conducted for testing purposes.
As one can see, even hundred-dimensional problems are solved in a few seconds.
Finding the support functions in this setting is trivial:

$$h_B(p) = \|p\|, \tag{17}$$

$$\nabla h_B(p) = \frac{p}{\|p\|}, \tag{18}$$

$$h_A(p) = \max_i (v_i, p), \tag{19}$$

$$\nabla h_A(p) = \arg\max_{v_i} (v_i, p). \tag{20}$$

2) Now let's move on to a more complicated series of tests. Let the set A be a convex hull of $n+1$ ellipsoids, and B be a ball. The results of the numerical experiments for dimension 100 is presented in Fig. 3.
In this case, it takes the algorithm tens of seconds to solve a 100-dimensional problem, which is about an order of magnitude longer than in the previous series. However, in most cases, it still took 5–7 iterations to converge. The increase in the runtime is due to the more complicated computation of the support function of A. Indeed, for an ellipsoid $E = v + MB_1(0)$, where

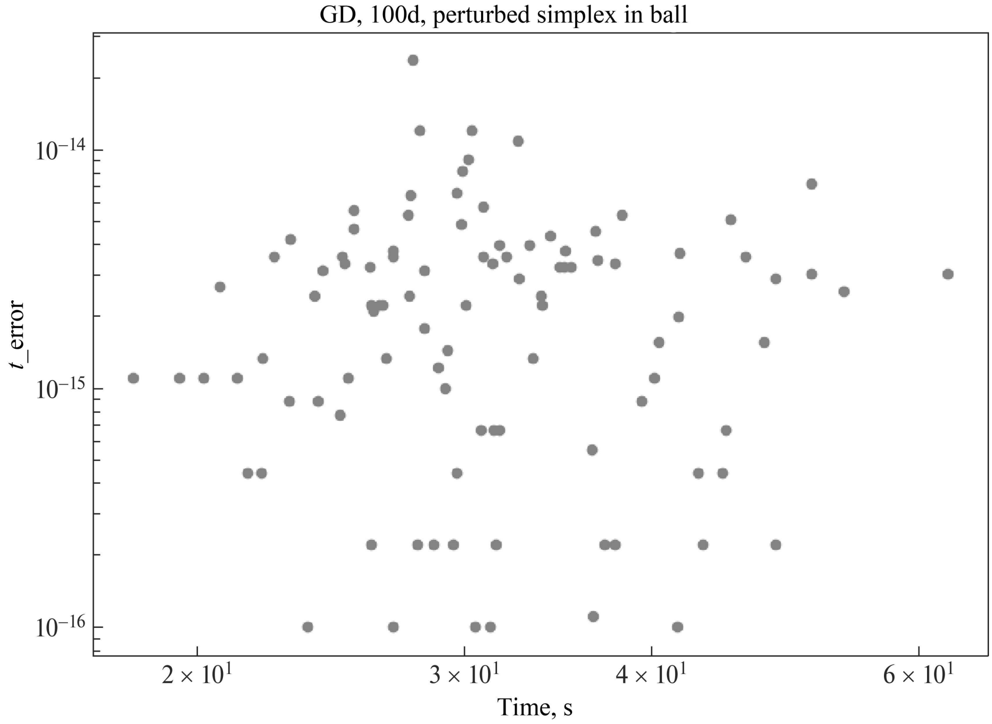

Fig. 3. Error of t versus runtime, $n = 100$.

M is a matrix, $v \in \mathbb{R}^n$,

$$h_E(p) = (p, v) + \max_{y \in MB_1(0)} (p, y) = (p, v) + \max_{u \in B_1(0)} (p, Mu) \tag{21}$$

$$= (p, v) + \max_{u \in B_1(0)} (M^T p, u) = (p, v) + \|M^T p\|, \tag{22}$$

$$\nabla h_E(p) = v + M \frac{M^T p}{\|M^T p\|} = v + \frac{MM^T p}{\|M^T p\|}. \tag{23}$$

If $A = conv(E_1 \cup \ldots \cup E_{n+1})$, $E_j = v_j + M_j B_1(0)$, one can compute the support function of A in the following way:

$$h_A(p) = \max_i h_{E_i}(p). \tag{24}$$

Then, the gradient of h_A is computed as a gradient of h_{E_k}, where E_k is the ellipsoid with the greatest $h_{E_k}(p)$.
Thus, calculating the support function of the convex hull on $n+1$ ellipsoids is significantly more computationally expensive.

4. SPECIAL CASE OF A SIMPLEX IN A BALL

Throughout this section, let B be a ball, and A be a simplex that contains the center of its circumscribed sphere. Clearly, in this case, $x^0 + t^0 \partial B$ coincides with the circumsphere of A. It is enough to consider the case $x^0 = 0$, $t^0 = 1$, $B = B_1(0)$, so the above equalities will be assumed throughout this chapter.

Lemma 3. *The minimums of $f_{t,x}$ depend only on x and do not depend on t.*

Definition 1. The set $F \subset \mathbb{R}^n$ is the closure of the set of such $x \in \mathbb{R}^n$, that $f_{t,x} = (x, p) + th_B(p) - h_A(p)$ has $n+1$ minimums on a unit sphere.

Since for $x \in F$ Algorithm 1 has a chance to converge (otherwise it will not be able to find $n+1$ minimums of $f_{t,x}$), we will study some properties of F, especially bounds on its volume.

Lemma 4. *If $n = 2$, then $F = conv(A \cup -A)$, $Vol(F) = 2Vol(A)$.*

Let $v_1, \ldots, v_{n+1}$ be the vertices of A. The following lemma describes F in terms of $v_1, \ldots, v_{n+1}$.

Lemma 5.

$$F = \{x \in \mathbb{R}^n | (v_j - x, v_j) \geqslant (v_j - x, v_i) \ \forall\, i, j\}, \tag{25}$$

$$F = \{x \in \mathbb{R}^n | (v_i - x, v_i - x) \geqslant (v_i - x, v_j - x) \ \forall\, i, j\}. \tag{26}$$

F is a polytope bounded by hyperplanes that contain v_i and are orthogonal to $v_i - v_j$, for all $i \neq j$.

Theorem 3. *Let $A \subset \mathbb{R}^n$ be an n-dimensional simplex with vertices $v_1, \ldots, v_{n+1}$, $F = \{x \in \mathbb{R}^n | (v_i - x, v_i - x) \geqslant (v_i - x, v_j - x) \ \forall\, i, j\}$. Then*

$$Vol(F) \geqslant n! Vol(A), \tag{27}$$

and if the equality holds, then F tiles $\mathbb{R}^n$.

Theorem 2 states that there is some neighborhood of the solution such that Algorithm 1 converges with some rate. However, it does not say anything about the size of this neighborhood. The following two theorems provide some information about this neighborhood.

Theorem 4. *Let $D = \max\limits_{i,j} \|v_i - v_j\|$. If at the current iteration Algorithm 1 is at t, x, where $x \in F$, and*

$$\|x\| \leqslant \frac{7}{16\sqrt{4 - D^2}}, \tag{28}$$

and $t \leqslant 1$, then $\|x'\| \leqslant \|x\|$, where t', x' is the state of the algorithm at the next iteration.

Theorem 5. *Let $d = \min\limits_{i,j} \|v_i - v_j\|$, and $d > 1$. Then for the state t, x, where $x \in int(F)$, $t \leqslant 1$, at the next iteration $\|x'\| < x$. Consequently, $x' \in F$.*

Since for large n, all edges random simplex inscribed in a unit sphere almost surely have a length greater than 1, the above theorem is almost surely applicable. In these cases, it is sufficient to have $\|x\| < 0.5$ at the initial iteration for Algorithm 1 to converge.

5. CONNECTING THE CHEBYSHEV CENTER OF A SIMPLEX AND A PERTURBED SIMPLEX

The previous section describes the behavior of the algorithm in the case of B being a ball, and A being a simplex that contains the center of its circumsphere. The solution to such a problem is obvious—one just needs to inscribe the simplex in a sphere. Therefore, so far the statements given in the previous chapter are useless in practice.

Theorem 2 describes a class of sets A such that their Chebyshev center can be found with Algorithm 1. Those sets are described as "perturbed simplices": $A = conv((v_1 + C_1) \cup \ldots \cup (v_{n+1} + C_{n+1}))$, where v_j are points, C_j are r-strongly convex sets, and f is guaranteed to have $n+1$ zeroes. One may hope that the behavior of Algorithm 1 for "perturbed" simplex should not be very different from its behavior for a simplex, which was studied in the previous section.

Let B be a ball throughout this section. Obviously, in Theorem 2 one can make $C_j \subset B_r(0)$ using translation, since C_j is r-strongly convex, therefore, contained in a ball of radius r. Let us state a theorem about the set F, defined in the previous section:

Theorem 6. *Let* $B = B_1(0)$*,* $A = conv((v_1 + C_1) \cup \ldots \cup (v_{n+1} + C_{n+1}))$*, and* C_j *are* r*-strongly convex, as in Theorem 2. Let* $C_j \subset B_r(0)$*.* F *is the set of* $x \in \mathbb{R}^n$ *such that* $f_{t,x}(p) = (x, p) - h_A(p)$ *has* $n+1$ *minimums on the unit sphere. Let* $\widetilde{A}$ *be a simplex with vertices* v_j*,* $\widetilde{A} = conv(v_1, \ldots, v_{n+1})$*. Let* $\widetilde{F}$ *be the set of* $x \in \mathbb{R}^n$*, such that* $f_{t,x}(p) = (x, p) - h_{\widetilde{A}}(p)$ *has* $n + 1$ *minimums on the unit sphere. Then*

$$F + B_{r/d}(0) \supset \widetilde{F}, \tag{29}$$

where $d = \min\limits_{i \neq j} \|v_i - v_j\|$*.*

Remark: it is easy to see that a slightly stronger statement holds,

$$F + B_\varepsilon(0) \supset \widetilde{F}, \tag{30}$$

where

$$\varepsilon = \frac{\max\limits_{i,j} \max\limits_{p \in S^{n-1}} (h_{C_i}(p) - h_{C_j}(p))}{d} \leqslant \frac{r}{d}. \tag{31}$$

For example, if all the sets C_j are the same, then $F \supset \widetilde{F}$.

The above theorem states that F, being the set of x such that Algorithm 1 can make it to the next iteration, is at least not much smaller for the problems with perturbed simplices.

6. CONCLUSION

We approach the optimization problem (1). Algorithm 1 that solves the problem for a specific class of sets A, B was suggested, implemented, and discussed. The algorithm performs several iterations. At each iteration, it searches minimums of the "gap" between A and $x + tB$ using gradient descent. In practice, the algorithm converges very fast even for 100-dimensional problems, showing a quadratic rate of convergence. However, to guarantee the presence of convergence, additional requirements on A and B have to be satisfied. For theoretical reasons, the behavior of the algorithm for the "simplex in a ball" problems was investigated. We introduced the set F of such initial conditions that there exist $n + 1$ minimums of $f_{t,x}$, and thus, Algorithm 1 can make it to the next iteration. Then, it was shown that if one slightly perturbs a simplex, F also changes slightly (Theorem 6). The main theoretical results are Theorem 1 about the convergence rate of the algorithm, Theorem 2 describing a class of problems that the algorithm is applicable to, Theorem 3 that bounds the volume of F and provides a nice result about tiling $\mathbb{R}^n$, Theorems 4, 5 that describe such initial conditions that the convergence of Algorithm 1 is guaranteed, and Theorem 6, which gives a connection between the set F when A is a simplex, and when A is a perturbed simplex.

7. ACKNOWLEDGEMENTS

The author is grateful to Maxim Balashov for setting the problem, providing useful literature, important discussions and text review. Also, I thank Dmitry Tsarev and Kseniia Petukhova for meaningful talks and support.

APPENDIX

Proof of Lemma 2. The functions (x, p) and $th_B(p)$ are differentiable. It is sufficient to prove that h_A is differentiable at the minimums of $f_{t,x}$.

Let p_0 be a minimum of $f_{t,x}$. h_A is a convex function, and it has a subdifferential ∂h_A. For the sake of contradiction, suppose $\partial h_A(p_0)$ has more than one element.

For any $\varepsilon > 0$ there exists δ such that for any $p \in B_\delta(p_0)$ the following inequality holds:

$$|\varphi(p) - \varphi(p_0) - (g, p - p_0)| < \varepsilon\|p - p_0\|. \tag{A.1}$$

For all $y \in \partial h_A(p_0)$,

$$h_A(p) \geqslant h_A(p_0) + (y, p - p_0), \tag{A.2}$$
$$-h_A(p) + h_A(p_0) \geqslant (-y, p - p_0). \tag{A.3}$$

The equations (A.1) and (A.3) give us

$$\varphi(p) - \varphi(p_0) - h_A(p) + h_A(p_0) \leqslant \varepsilon\|p - p_0\| + (g, p - p_0) - (y, p - p_0), \tag{A.4}$$
$$f(p) - f(p_0) \leqslant \varepsilon\|p - p_0\| + (g - y, p - p_0). \tag{A.5}$$

Since $\partial h_A(p_0)$ has more than one element, there is such $y \in \partial h_A(p_0)$ that $y \neq g$. Then, at $p - p_0 = \delta\frac{y-g}{\|y-g\|}$,

$$f(p) - f(p_0) \leqslant \varepsilon\delta - \delta\|g - y\|, \tag{A.6}$$

which is negative for sufficiently small ε. This contradicts the fact that f has a minimum at p_0.

Proof of Theorem 1. It is sufficient to prove the case $x^0 = 0$ because other cases can be reduced to this by a suitable translation. We will assume that $x^0 = 0$. For p from some neighborhood $U(p_j^0)$ the following inequalities hold:

$$\begin{cases} f_{t,x}(p) = (x, p) + th_B(p) - h_A(p) \geqslant (x, p) + (t - t^0)h_B(p) + \mu\|p - p^0\|^2, \\ f_{t,x}(p) = (x, p) + th_B(p) - h_A(p) \leqslant (x, p) + (t - t^0)h_B(p) + M\|p - p^0\|^2. \end{cases} \tag{A.7}$$

Let p' be a minimum of $f_{t,x}$ in $U(p_j^0)$. First, let us derive a bound for $\|p' - p_j^0\|$.

$$\begin{cases} f_{t,x}(p') = (x, p') + th_B(p') - h_A(p') \geqslant (x, p') + (t - t^0)h_B(p') + \mu\|p' - p^0\|^2, \\ f_{t,x}(p') \leqslant \min\limits_{p \in U(p_j^0)} \{(x, p) + (t - t^0)h_B(p) + M\|p - p^0\|^2\} \leqslant (x, p_j^0) + (t - t^0)h_B(p_j^0). \end{cases} \tag{A.8}$$

Then,

$$(x, p') + (t - t^0)h_B(p') + \mu\|p' - p^0\|^2 \leqslant f_{t,x}(p') \leqslant (x, p_j^0) + (t - t^0)h_B(p_j^0), \tag{A.9}$$
$$(x, p') + (t - t^0)h_B(p') + \mu\|p' - p^0\|^2 \leqslant (x, p_j^0) + (t - t^0)h_B(p_j^0). \tag{A.10}$$

Which gives us

$$\mu\|p' - p^0\|^2 \leqslant (x, p_j^0 - p') + (t - t^0)(h_B(p_j^0) - h_B(p')). \tag{A.11}$$

Using the Cauchy–Schwarz inequality and the definition of the gradient of a support function:

$$\mu\|p' - p^0\|^2 \leqslant (x, p_j^0 - p') + (t - t^0)(h_B(p_j^0) - h_B(p')) \tag{A.12}$$
$$\leqslant \|x\|\|p' - p_j^0\| + |t - t^0|\|\nabla h_B(p_j^0)\|\|p' - p_j^0\|(1 + o(1)). \tag{A.13}$$

Then,

$$\|p' - p_j^0\| \leqslant Const(\|x\| + |t - t^0|) \tag{A.14}$$

for some constant $Const$, which does not depend on x and t in some neighborhood of (t^0, x^0). Now we have a bound on the error $\|p' - p_j^0\|$.

For the state of the algorithm on the next iteration, t', x',

$$(x', p'_j) + t' h_B(p'_j) - h_A(p'_j) = 0 \ \forall j \in \{1, \dots, n+1\}, \tag{A.15}$$

$$(x', p'_j) + (t' - t^0) h_B(p'_j) = h_A(p'_j) - t^0 h_B(p'_j). \tag{A.16}$$

This is a system of linear equations for x' and $t' - t^0$. Since for any fixed j, the right-hand side of the equation is $h_A(p'_j) - t^0 h_B(p'_j) = -f(p'_j)$, the norm of the right-hand side in the entire system is not greater than $\sqrt{n+1} M \|p'_j - p_j^0\|^\alpha \leqslant Const(\|x\| + |t - t^0|)^\alpha$. The matrix of this system is not degenerate if $0 \in conv(\{p'_1, \dots, p'_{n+1}\})$, which holds true if t and x are close enough to t^0, x^0. The inverse matrix has a bounded norm for $(p'_1, \dots, p'_{n+1})$ from some neighborhood of $(p_1^0, \dots, p_{n+1}^0)$. Thus, the following bound holds:

$$\|(t' - t^0, x'_1 - x_1^0, \dots, x'_n - x_n^0)^T\| \leqslant Const(\|x\| + |t - t^0|)^\alpha \tag{A.17}$$

$$\leqslant Const\|(t - t^0, x_1 - x_1^0, \dots, x_n - x_n^0)^T\|^\alpha, \tag{A.18}$$

which completes the proof.

Proof of Theorem 2. It is sufficient to prove the statement for $x^0 = 0$, $t^0 = 1$.

1) Notice that $(A \cap \partial B) \subset \bigcup_j (v_j + C_j)$.
   Indeed, by Caratheodori's theorem, A is a union of simplices with vertices in sets $v_j + C_j$. A simplex which is a subset of a ball can intersect the boundary of the ball only by its vertices. So, every point from $A \cap \partial B$ belongs to $v_j + C_j$ for some j.
2) For each j, $(v_j + C_j) \cap \partial B$ has at most one element.
   By contradiction: if for some j there are distinct $a, b \in (v_j + C_j) \cap \partial B$. Let M be the r-strongly convex hull of $\{a, b\}$: $M = strconv_r(\{a, b\})$. Since the sets $v_j + C_j$ are strongly convex,
   $$M \subset v_j + C_j. \tag{A.19}$$
   Since $a, b \in \partial B = \partial B_1(0)$, and $r < 1$,
   $$M \cap (\mathbb{R}^n \setminus B) \neq \emptyset. \tag{A.20}$$
   Then,
   $$(v_j + C_j) \cap (\mathbb{R}^n \setminus B) \neq \emptyset, \tag{A.21}$$
   which contradicts the fact that $(v_j + C_j) \subset B$.
3) For each j, $(v_j + C_j) \cap \partial B$ has exactly one element.
   By the previous item, $v_j + C_j$ intersects ∂B by at most one point. By the condition of the theorem, f has at least $n+1$ zeros, then $A \cap \partial B$ has at least $n+1$ elements. So $(v_1 + C_1) \cup \ldots \cup (v_{n+1} + C_{n+1})$ intersects ∂B by at least $n+1$ points. Thus, each set $v_j + C_j$ intersects ∂B.
4) In some neighborhood of each zero of f, f has a quadratic upper bound.
   Let $p_1^0, \dots, p_{n+1}^0$ be zeroes of f. For the simplex $S = conv(\{p_1^0, \dots, p_{n+1}^0\})$ and a point p from some neighborhood of p_j^0 the following equality holds: $h_S(p) = (p, p_j^0) = 1 - \frac{1}{2}\|p - p_j^0\|^2$. Since $S \subset A$, $h_A(p) \geqslant h_S(p)$, which gives us the desired bound $1 - h_A(p) \leqslant 1 - h_S(p) = \frac{1}{2}\|p - p_j^0\|^2$.
5) In some neighborhood of each zero of f, f has a quadratic lower bound.
   $v_j + C_j \subset B_r((1-r)p_j^0)$, consequently, $A \supset conv(B_r((1-r)p_1^0) \cup \ldots \cup B_r((1-r)p_{n+1}^0) = B_r(0) + (1-r)conv(\{p_1^0, \dots, p_{n+1}^0\})$. If A is a sum of a simplex and a ball, the desired bound holds, then for the given A the bound holds as well.

We have shown that all the conditions of Theorem 1 hold, with $\alpha = 2$. Then, the quadratic convergence of Algorithm 1 is present in this problem.

Proof of Theorem 3. Define a lattice Λ, spanned by $v_i - v_{n+1}$, $1 \leqslant i \leqslant n$:

$$\Lambda = \{c_1(v_1 - v_{n+1}) + \ldots + c_n(v_n - v_{n+1}) | c_i \in \mathbb{Z}\}. \tag{A.22}$$

The volume of the unit cell of this lattice equals $n! Vol(A)$.

Consider the Voronoi diagram for Λ. Let C be the cell of the Voronoi diagram that contains zero.

$$C = \left\{ x \in \mathbb{R}^n | (x, y) \leqslant \frac{1}{2}(y, y) \; \forall \, y \in \Lambda \setminus \{0\} \right\}. \tag{A.23}$$

It is easy to see that

$$F = \left\{ x \in \mathbb{R}^n | (x, v_i - v_j) \leqslant \frac{1}{2}(v_i - v_j, v_i - v_j) \; \forall \, i, j \right\}. \tag{A.24}$$

Since $v_i - v_j \in \Lambda$, the set of linear constraints defining F is a subset of the set of linear constraints defining C. Then $C \subset F$, and

$$Vol(F) \geqslant n! Vol(A) \tag{A.25}$$

follows immediately.

If $Vol(F) = Vol(C) = n! Vol(A)$, then $F = C$, and F tiles $\mathbb{R}^n$, because C tiles $\mathbb{R}^n$.

Proof of Theorem 6.

1) If for each j there exists such $p_j \in S^{n-1}$ that for any $i \neq j$

$$h_{C_j}(p_j) + (v_j, p_j) \geqslant h_{C_i}(p_j) + (v_i, p_j), \tag{A.26}$$

then $x \in F$. Indeed, each minimum of $f_{t,x}(p)$ corresponds to its own C_j (see Theorem 2, item 3), and in order for C_j to have a minimum associated with it, it is enough to have $(x, p_j) - h_{v_j + C_j}(p_j) \leqslant (x, p_j) - h_{v_i + C_i}(p_j)$ for some p_j, which is equivalent to (A.26).

2) Consider the previous item for $p_j = \frac{v_j - x}{\|v_j - x\|}$. Then if for each j and any $i \neq j$

$$h_{C_j}(v_j - x) + (v_j, v_j - x) \geqslant h_{C_i}(v_j - x) + (v_i, v_j - x), \tag{A.27}$$

then $x \in F$.

3)

$$0 \leqslant h_{C_j}(p) \leqslant r, \tag{A.28}$$

since $C_j \subset B_r(0)$. This gives us

4) If for each j and $i \neq j$

$$(v_j, v_j - x) \geqslant (v_i, v_j - x) + r, \tag{A.29}$$

then $x \in F$.

5) Compare:

$$\{x \in \mathbb{R}^n \mid (v_i - v_j, x) - (v_i - v_j, v_j) \geqslant r\} \subset F, \tag{A.30}$$

$$\{x \in \mathbb{R}^n \mid (v_i - v_j, x) - (v_i - v_j, v_j) \geqslant 0\} = \widetilde{F}. \tag{A.31}$$

If $x \in \widetilde{F}$, then $x + r\frac{v_i - v_j}{\|v_i - v_j\|^2} \in F$. So,

$$F + B_{r/d}(0) \supset \widetilde{F}, \tag{A.32}$$

as claimed.

REFERENCES

1. Balashov, M.V., Covering a Set by a Convex Compactum: Error Estimates and Computation, *Math. Notes*, 2022, vol. 112, pp. 349–359. https://doi.org/10.1134/S0001434622090024
2. Polyak, B.T., *Vvedenie v optimizatsiyu* (Introduction to Optimization), Moscow: Nauka, 1983.
3. Tyrrell Rockafellar, R., *Convex Analysis*, Princeton University Press, 1997.
4. Botkin, N. and Turova, V., An algorithm for finding the Chebyshev center of a convex polyhedron, *Appl. Mat. Optim.*, 1995, vol. 29, pp. 211–222.
5. Xia, Y. and Yang, M., Chebyshev center of the intersection of balls: complexity, relaxation and approximation, *Mathematical Programming*, 2021, vol. 187, pp. 287–315.
6. Frankowska, H. and Olech, C., R-convexity of the integral of set-valued functions, *Contribut. Anal. Geometry*, 1980, vol. 117–129.
7. Vial, J.-Ph., Strong and Weak Convexity of Sets and Functions, *Mathematics of Operations Research*, 1983, vol. 8, no. 2, pp. 231–259.
8. Boyd, S. and Vandenberghe, L., *Convex optimization*, Cambridge University Press, 2004.
9. Li, X., Zhu, Z., Man-Cho, S., and Lee, J.D., Incremental Methods for Weakly Convex Optimization, *arXiv*, 2019, vol. 1907.11687v1.
10. Davis, D., Drusvyatskiy, D., MacPhee, K.J., and Paquette, C., Subgradient Methods for Sharp Weakly Convex Functions, *arXiv*, 2018, vol. 1803.02461v1.
11. Schneider, R. and Uschmajew, A., Convergence results for projected line search methods on varieties of low-rank matricies via Lojasiewicz inequality, *SIAM J. Optim.*, 2015, vol. 25, no. 1, pp. 622–646.
12. Balashov, M.V., About the Gradient Projection Algorithm for a Strongly Convex Function and a Proximally Smooth Set, *J. Convex Anal.*, 2017, vol. 24, no. 2, pp. 493–500.
13. Bello-Cruz, Y., Li, G., and Nghia, T.T.A., On the Linear Convergence of Forward–Backward Splitting Method: Part I – Convergence Analysis, *J. Optim. Theory Appl.*, 2021, vol. 188, pp. 37–401.
14. Ioffe, A.D., Metric regularity – a survey Part I, *J. Austral. Math. Soc.*, 2016, vol. 101, pp. 1–56.
15. Absil, P.-A., Mahony, R., and Sepulchre, R., *Optimization Algorithms on Matrix Manifolds*, Princeton University Press, 2008.
16. Cen, X., Xia, Y., Gao, R., and Yang, T., On Chebyshev Center of the Intersection of Two Ellipsoids, *Optimization of Complex Systems: Theory, Models, Algorithms and Applications*, Springer International Publishing, 2020, pp. 135–144.
17. Beltran, F., Finardi, E.C., Fredo, G.M., and Oliveira, W., Improving the performance of the stochastic dual dynamic programming algorithm using Chebyshev centers, *Optim. Engineer.*, 2022, vol. 23, pp. 147–168.
18. Beck, A. and Eldar, Y.C., Regularization in Regression with Bounded Noise: A Chebyshev Center Approach, *SIAM J. Matrix Anal. Appl.*, 2007, vol. 29, no. 2, pp. 606–625.
19. Cerone, V., Piga, D., and Regruto, D., Set-Membership Error-in-Variables Identification Through Convex Relaxation Techniques, *IEEE Transact. Autom. Control*, 2012, vol. 57, no. 2, pp. 517–522.
20. Hou, J., Teng, F., Yin, W., Song, Y., and Hou, Y., A Cost-Effective Cyber-Defense Strategy: Attack-Induced Region Minimization and Cybersecurity Margin Maximization, *arXiv*, 2023, vol. 2302.07597.
21. Samadi, S., Roux, J., Tanguy, A., Caron, S., and Kheddar, A., Some journal publication in English, *IEEE Robot. Autom. Lett.*, 2021, vol. 6, no. 2, pp. 4032–4039.
22. Ren, X., Mo, Y., Chen, J., and Johansson, K.H., Secure state estimation with byzantine sensors: A probabilistic approach, *IEEE Transact. Autom. Control*, 2020, vol. 65, no. 9, pp. 3742–3757.

This paper was recommended for publication by P.S. Shcherbakov, a member of the Editorial Board