Московский физико-технический институт
(национальный исследовательский университет)
Физтех-школа прикладной математики и информатики
Кафедра интеллектуальных систем

Латыпов Ильгам Магданович

Balancing Efficiency and Interpretability: A New Approach to Multi-Objective Optimization with High Computation Costs in Lipschitz Functions

03.03.01 — Прикладные математика и физика

Выпускная квалификационная работа бакалавра

Научный руководитель:
к.т.н. Ю. В. Дорн

Москва — 2024

Аннотация

In practical engineering and optimization scenarios, tackling multi-objective optimization (MOO) problems often involves employing scalarization methods. Traditional approaches, while being effective, often entail significant computational overhead due to iterative computations. In this paper, we propose a novel method specifically tailored for Lipschitz objective functions aimed at computing function values only once, thus substantially reducing computational costs. This approach is advantageous in scenarios where function computation is expensive or where function values are computed once and then computation become unavailable. Our algorithm uses the Gembchiki[9] scalarization method, is formulated as a convex optimization problem and offers efficient and practical solution to MOO problems. Additionally, we propose interpretable parameters selection for the scalarization method, enhancing the interpretability and practical applicability of the optimization results. Empirical evaluations, conducted on graph-based supply network problems, demonstrate the effectiveness and scalability of our approach, highlighting its potential to address computational challenges in MOO across various domains.

1 Introduction

Applied problems are often formulated in a way that lacks a single optimal solution, requiring consideration of multiple criteria. Multiple Criteria Decision Making (MCDM) [11] is commonly employed to address such challenges. Tasks within this domain, along with the methods used to tackle them, find widespread applications in various fields such as: supply network and telecommunication network parameter selection [1, 4, 7, 16], machine learning [21, 22, 24], chemistry [20], biology [5] and engineering [14]

In multiobjective optimization, there are two predominant approaches: approximating the Pareto front and metric scalarization, which involves reducing the multi-criteria problem to a single-criteria problem.

A multitude of methods have been devised for reconstructing the Pareto front, including genetic and evolutionary approaches [12, 19], interactive approaches [18], Bayesian-based methods [6, 23], and methods based on reconstructing descent directions of functions [8].

Methods for reducing the problem to a single-objective optimization are also widespread. Popular ones include weighting methods, the weighted t-th power, the weighted quadratic problem, and the ϵ-constraint approach. Families of methods with a target point, such as distance-function-based approaches and the achievement function approach, and direction-based approaches like the Pascoletti and Serafini approach, the reference direction approach, and the goal-attainment approach, are thoroughly discussed in [10]. Additionally, several books are dedicated to multi-criteria optimization [10, 11, 17].

Upon scrutiny of scalarization methods, it becomes clear that all methods necessitate the introduction of parameters such as weights for functions weighting. In this work, we propose a method for finding an approximate solution for task formulation, that often appears in practice. It aimed for Lipschitz functions under severely constrained computational budgets: we can use only some precomputed values to make a decision. We also note the connectivity of scalarization that we use with one stated at [9].

The subsequent text is structured as follows: In Section 2, the mathematical problem statement and method are introduced, highlighting the connection with the previous formulation and justifying the relevance of our approach. Section **??** outlines the proposed optimization method.

2 Problem statement

Multiobjective optimization problem is formulated as follows:

$$\min_x f \triangleq (f_1(x), ..., f_m(x))^T \qquad (T_0)$$
$$\text{s.t. } x \in K$$

There $x \in \mathbb{R}^n$. Optimization functions $f_i : \mathbb{R}^n \to \mathbb{R} \;\; i = \overline{1, m}$. Decision set K is convex, nontrivial, compact and equipped with a norm $\|\cdot\|$. For example, one can consider Euclidean space with linear constraints: $K = \{x \in \mathbb{R}^n : \;\; Ax \leq b\}$. It remains to define what means optimality for vector of solution. The common way is to use Pareto optimality, but our goal is to get one solution, that will be good in our practice view. We suppose that functions are positive valued. Production costs of products can be considered as such metrics.

We wish to develop a method for practical implementation. To this end, we need to consider how practical tasks are defined, which often involve setting requirements that the resulting solution must be no worse or better than the current solution by a specified percentage. This same percentage is applied to several indicators. Let v_i represent the value of function f_i at point $x_i : f_i(x_i) = v_i$. In this case, the above is formally written as follows:

$$1 - \gamma \leq \frac{f(x)}{f_i(x_i)} \leq 1 + \gamma$$

Where γ is a measure of function change. Depending on the optimization goal for a given function, one of the constraints may be removed – for example, for the maximization of a function, only the restriction from below is relevant. Based on this constraint, we formulate the optimization problem as follows:

$$\min_{x,\gamma} \gamma \qquad (T_1)$$
$$\text{s.t. } x \in K$$
$$|f_i(x) - v_i| \le \gamma v_i \;\; \text{for} i \in \overline{1,m}$$

That is, we aim to find x and γ, such that the change in each function value is at most within the specified interval given by γ. Let's give an example to clarify the problem statement

Пример. *Function f_i is a loss function, computed on a model with parameters x. x_i is an optimal parameters for this loss function. Then solution of task T_1 is a parameter, that spoils metrics less than with factor γ. Note that in this case x_i is hardly computable point.*

Пример. *e The delivery service have been working with different strategies – points x_i and compute metrics – $f_i(x_i)$. Now they want to set some parameters, that will guarantee the values of the metrics if the situation will be similar to the ones already known.*

As we note earlier, scalarization method proposed by [9] is similar to our problem statement. They introduce optimization problem as:

$$\min_{x,\gamma} \gamma \qquad (T_1)$$
$$\text{s.t. } x \in K$$
$$f_i(x) - w_i\gamma \le f_i^* \;\; \text{for} \;\; i \in \overline{1,m}$$

f_i^* interpreted as a desired level of performance of corresponding index. Values w_i denote relative importance of i-th functions change. if we choose $w_i = f_i^* = v_i$ we get optimization problem

$$\min_{x,\gamma} \gamma \qquad (\overline{T_1})$$
$$\text{s.t. } x \in K$$
$$f_i(x) \le (\gamma + 1)v_i \;\; \text{for} \;\; i \in \overline{1,m}$$

Which is similar to our approach.

Определение 1. *Function* $f : \mathbb{R}^n \to \mathbb{R}+$ *is satisfy Lipschitz condition on* K *with norm* $\| \cdot \|$*:*

$$\exists L > 0 : \ \forall x, y \in K : \ |f(x) - f(y)| \leq L\|x - y\|$$

Now we are ready to present proposed method for optimization problem in case, when one can not search optimal point iteratively. This can happen for various reasons – getting x_i is expensive, v_i computation is expensive or even impossible 2. There is no gradient operation available for iteration over it. We have only precomputed metrics and knowledge that parameters are Lipschitz w.r.t. parameter. Why we think, that metrics can be Lipschitz and hardly computable? There is an example from network optimization, modern tasks from this field often take large sizes and can be expressed in the form of linear optimization problems [2, 15]. According to [13] problems of linear programming type are satisfy Lipschitz conditions with respect to changes in the right-hand side data of the problem. We provide an example of such problem in section 5

3 Method

Let's put all the symbols together: for all $i = \overline{1, m}$ there are metrics $f_i(x) : \mathbb{R}^n \to \mathbb{R}$ to be minimized. It satisfies Lipschitz condition 1 with Lipschitz constant L_i. And computed at $x_i : f_i(x_i) = v_i$. We search $x \in K$ that is feasible for T_1 with some γ. As we actually mentioned, method should not compute functions while solution search. We can only take advantage of Lipschitzness. Consider some $x \in K$ and γ that satisfy

$$\|x_i - x\| \leq \frac{\gamma v_i}{L_i} \Rightarrow v_i - f(x) \leq |v_i - f(x)| \leq \gamma v_i \tag{1}$$

Which means, that x is feasible for this γ. We can define new optimization problem:

$$\min_{x,\gamma} \gamma \tag{T_2}$$
$$\text{s.t. } x \in K$$
$$\|x - x_i\| \le \frac{1}{L_i}\gamma v_i \quad \forall i \in \overline{1,m}$$

This problem is convex and do not compute functions. It is approximation and so it will return value worse than exact solution of problem T_1. We can ease constraints, if we have information about the monotony of the function by parameter and the target change of this function is one-sided. For convenience, we will assume that it needs to be minimized. Monotony is found, for example, in a linear programming problem – the function is monotonous in right-hand parameters, since increasing them makes the constraints more soft. For this case, we introduce $\texttt{clip} : \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}^n$ operator:

$$\texttt{clip}(x, y, f)_i = \begin{cases} \max(x_i - y_i, 0) & f \text{ increase on } i\text{-th parameter} \\ \min(y_i - x_i, 0) & f \text{ decrease on } i\text{-th parameter} \\ x_i - y_i & \text{none of the above or not known} \end{cases} \tag{2}$$

Now we can rewrite optimization problem as and get proposed reformulation of optimization problem:

$$\min_{x,\gamma} \gamma \tag{T_3}$$
$$\text{s.t. } x \in K$$
$$\|clip(x,x_i, f_i)\| \le \frac{1}{L_i}\gamma v_i \quad \forall i \in \overline{1,m}$$

Here is an example to demonstrate which vector the operator $\texttt{clip}(x,y,f)$ returns 1.

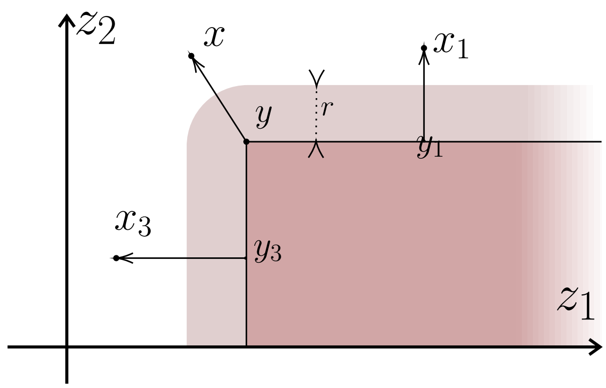

Рис. 1: Let the function f decrease in z_1 and increase in z_2 and it to be minimized. The y point is selected. In this case, the operator returns 0 for points in the gray area. For the remaining points, the operator returns the vector shown in the figure. In fact, the projection is done coordinate-wise. The darkened area shows an area of space, the projection from which will have a norm of no more than the r indicated in the figure.

4 Theorems and proofs

now let's examine the properties of the stated optimization problems. It is convex. We show that the solution of the T_3 problem satisfies the necessary conditions, then we consider how the solution is violated if Lipschitz constants are not known accurately, which often occurs in practice due to lack of access to the function.

Теорема 1. *Solution of T_3 satisfy T_1*

Доказательство. Let $z_i = \texttt{clip}(x, x_i, f_i)$ and $y_i = x_i - z_i$. For it we have $f_i(y_i) \le f_i(x_i)$ and using that $\|x_i - y_i\| = \|z_i\| \le \frac{\gamma v_i}{L_i}$ get $f_i(x) - f_i(y_i) \le \gamma v_i$. Summing up the two expressions obtained, we get $f_i(x) - f_i(x_i) \le \gamma v_i$. $\square$

Теорема 2. *Let γ^* be a solution of T_3. If we have approximations of the Lipschitz constants $\widetilde{L}_i = \kappa_i L_i$, then by solving T_3, in which the Lipschitz constants are changed to $\widetilde{L}_i$ we obtain x, for which $|f(x) - f(x_i)| \le \frac{\kappa_{\max}}{\kappa_i}\gamma^* v_i$. Here, $\kappa_{\max} = \max_{i=\overline{1,m}} \kappa_i$.*

Доказательство. let's denote the problem with inaccurate Lipschitz constants introduced in the theorem as $\widetilde{T3}$ and its solution as $\widetilde{\gamma}$. And actually we introduce one more problem, in which L_i is changed to $\kappa_{\max} L_i$. Denote the third problem as $\overline{T3}$ and its solution as $\overline{\gamma}$. We can see, that T_3 and $\overline{T3}$ actually the same problem, and for them $\gamma^* = \frac{\overline{\gamma}}{\kappa_{\max}}$.

Also, for problems $\widetilde{T3}$ and $\overline{T3}$ the relation $\widetilde{\gamma} \leq \overline{\gamma}$ holds: in the $\widetilde{T3}$, the constraints are relaxed constraints of $\overline{T3}$, and so solution of $\overline{T3}$ is a feasible point.

Therefore, we get $\widetilde{\gamma} \leq \overline{\gamma} \leq \gamma^* \kappa_{\max}$. And so for solution of $\widetilde{T3}$:

$$f(x) - f(x_i) \leq \frac{L_i}{L_i \kappa_i} (\widetilde{\gamma} v_i) \leq \frac{\kappa_{\max}}{\kappa_i} (\overline{\gamma} v_i) \tag{3}$$

$\square$

This theorem shows that the parameters need to be evaluated approximately the same way.

Комментарий 1. *Optimality of the obtained approximate solution.*

To estimate the deterioration of the solution, consider an example with two functions $f_1(x) = 1 + L_1|x|$ *and* $f_2(x) = 1 + L_2|x-1|$ *and take* $x_1 = 0, x_2 = 1$

Then for T_3 *we get the solution* $\gamma^* : (\gamma^* - 1)(\frac{1}{L_1} + \frac{1}{L_2}) = 1$. *For the same problem with inaccurate constants* $\widetilde{L}_i = \kappa_i L_i$, *we get* $\widetilde{\gamma} : (\widetilde{\gamma} - 1)(\frac{1}{L_1 \kappa_1} + \frac{1}{L_2 \kappa_2}) = 1$.

From here

$$\frac{\widetilde{\gamma} - 1}{\gamma^* - 1} = \frac{\frac{1}{L_1} + \frac{1}{L_2}}{\frac{1}{L_1 \kappa_1} + \frac{1}{L_2 \kappa_2}} = \frac{\frac{L_2}{L_1} + 1}{\frac{L_2}{L_1 \kappa_1} + \frac{1}{\kappa_2}} \approx \kappa_2 \tag{4}$$

For the case of $L_2 << L_1$ *and* $\kappa_1 = 1$ *(we know exactly* L_1*). The resulting solution will be close to the boundary obtained in the theorem.*

5 Numerical experiments

To demonstrate the effectiveness of the proposed method, we conduct numerical experiments. The first Next, we investigate the minimum cost concurrent flow problem, where the objective functions f_i share a common structure but differ in parameters, representing various scenarios of network congestion. In the third experiment, we explore different metrics applied to graph-based optimization problems. These experiments collectively showcase the versatility and applicability of our approach across diverse optimization scenarios.

5.1 Production

Сценарий эксперимента основан на классической задаче линейного программирования. Пусть дана компания. Ее работу можно разделить на периоды. The company produces k types of products with amounts $x \in \mathbb{R}^k$ and realize it with cost c. For this it needs n types of resources with amounts $b \in \mathbb{R}^n$, which are purchased at a cost c_b at the beginning of the period. Some of the resources may run out during the period, so the company buys additional resources y at the cost $c_a \geq c_b$ componentwise. For the product i it uses a_{ij} amount of resource j. Denote it $A = \|a_{i,j}\|_{i,j=1} \in \mathbb{R}^{n \times k}$.

Для упрощения примера считаем, что компания закупает недостающие ресурсы ровно столько сколько нужно в течение периода.

В нашей постановке x может быть случайным – это то, сколько компания смогла продать в течение периода. С учетом всех трат, доход компании за период записывается:

$$
\begin{aligned}
f(b, x, c, c_b, c_a) &= c^T x - c_b^T b - c_a^T y \\
y &= \max(0, Ax - b)
\end{aligned}
$$

Эта функция Липшицева по параметру b с константой Липшица $\max(\mathrm{abs}(c_b), \mathrm{abs}($ $c_b))$ в норме $\|\cdot\|_1$.

The company has worked for m periods and observed for $i \in \overline{1, m} : c_i, x_i, c_b^i, c_a^i$ for resource amounts b_i. It is necessary to find a strategy for purchasing resources that will give good enough income if situation will look as one of observed situations. Besides that there is a constraint for budget: $K = \{b : c_b^T b \leq B\}$.

We vary the budget in given range and get the experimental results. We provide two experiments: first is about using different metrics and the second about calculation on large scale problems.

5.1.1 Production: comparison

Для этого эксперимента рассматриваем number of cases $m = 50$, number of types of resources $n = 80$, number of types of products $k = 30$. and vary budget B. For approximate solution with $\|\cdot\|_1$ norm we use exact constants. Actually we compare it with solution with $\|\cdot\|_2$ norm due to the equivalence of norms. In this case we use the same Lipschitz constants, but they are perturbed by norm change. And for clarity we compare them with exact solution.

На рисунке 2 нарисованы точки, полученные при разных бюджетах B. Сверху вниз изображены: приближенное решение, полученное использованием $\|\cdot\|_1$ нормы; приближенное решение, полученное использованием $\|\cdot\|_2$ нормы. Точное решение. Вертикальная линия – минимальный бюджет, использованный в точках b_i. На осях отображаются бюджет; среднее $\pm$ дисперсия для отношения функций $f_i(b)/f_i(b_i)$ для кейсов и полученные $1-\gamma$. Так как мы ищем точку, в которой $1-\gamma \leq f_i(b)/f_i(b_i)$, то при росте бюджета и уменьшении γ ожидается увидеть рост отношения значений функций.

На графиках видим, что найденные γ для приближенных решений не получаются достаточно информативными – они могут быть и больше 1 из-за домножения на число, что и случилось для обеих приближенных решений. Однако как показывает график посередине, найденная точка не сильно уступает точному решению в качестве в среднем. Для аппроксимации через $\|\cdot\|_2$ норму алгоритм при больших бюджетах начинает брать слишком большие значения b, что приводит к лишним затратам. $\|\cdot\|_1$ норма плохо работает, что может быть объяснено тем, что при использовании такой нормы находятся разреженные решения, что не подходит для данной задачи – для производства нужны все ресурсы.

На 3 нарисовано отношение средних графиков, нарисованных на 2. А именно, из результатов точного решения вычитается приближенное решение и это нормируется на результаты точного решения. Видим, что решение, полученное $L1$ нормой сильно уступает точному решению, в то время как решение, полученное L_2 нормой не отстает и в некоторых случаях в среднем даже лучше точного решения. Однако на больших бюджетах решение через первую норму лучше чем L_2 решение.

5.1.2 Production: ablation study

Чтобы сравнить производительность алгоритмов на больших задачах, рассмотрим $m = 50; \quad n = 600; \quad k = 300$. Алгоритмы реализованы на cvxpy. Для поиска приближенного решения во второй норме используется solver ECOS, для точного решения SCIPY, так как он показал лучшую производительность на наших экспериментах. В эксперименте для заданного бюджета запускаем вычисления 11 раз и измеряем время вычисления. Результат эксперимента представлен на 4. Алгоритм на протестированных параметрах работает до 4 раз быстрее.

5.2 Minimal cost concurrent flow

We are given a graph $\mathcal{G}(V, E)$, with m nodes and n edges. Edges $e \in E$ is assigned with a capacity b_e and a flow cost c_e. Besides that there are demands $d_i = (s_i, t_i, f_i)$, which is a triplet sourse s, target t and flow amount f. We denote A correspondency matrix of

problem minimal cost concurrent flow is formulated as follows: We are given a graph $\mathcal{G}(V, E)$, with m nodes and n edges. Edges $e \in E$ is assigned with a capacity b_e and a flow cost c_e. Besides that there are demands $d_i = (s_i, t_i, f_i)$, which is a triplet sourse s, target t and flow amount f. Let us denote demands as a correspondency matrix D, where $\forall i : D_{s_i, t_i} = f_i$. Our goal is to spread flows in graph to get the minimal cost of delivery. It could be written as an optimization problem as follows []todoдобавить источники:

$$\min_F c^T F \mathbf{1}$$

$$\text{s.t. } F\mathbf{1} \leq b$$

$$NF = -L_D$$

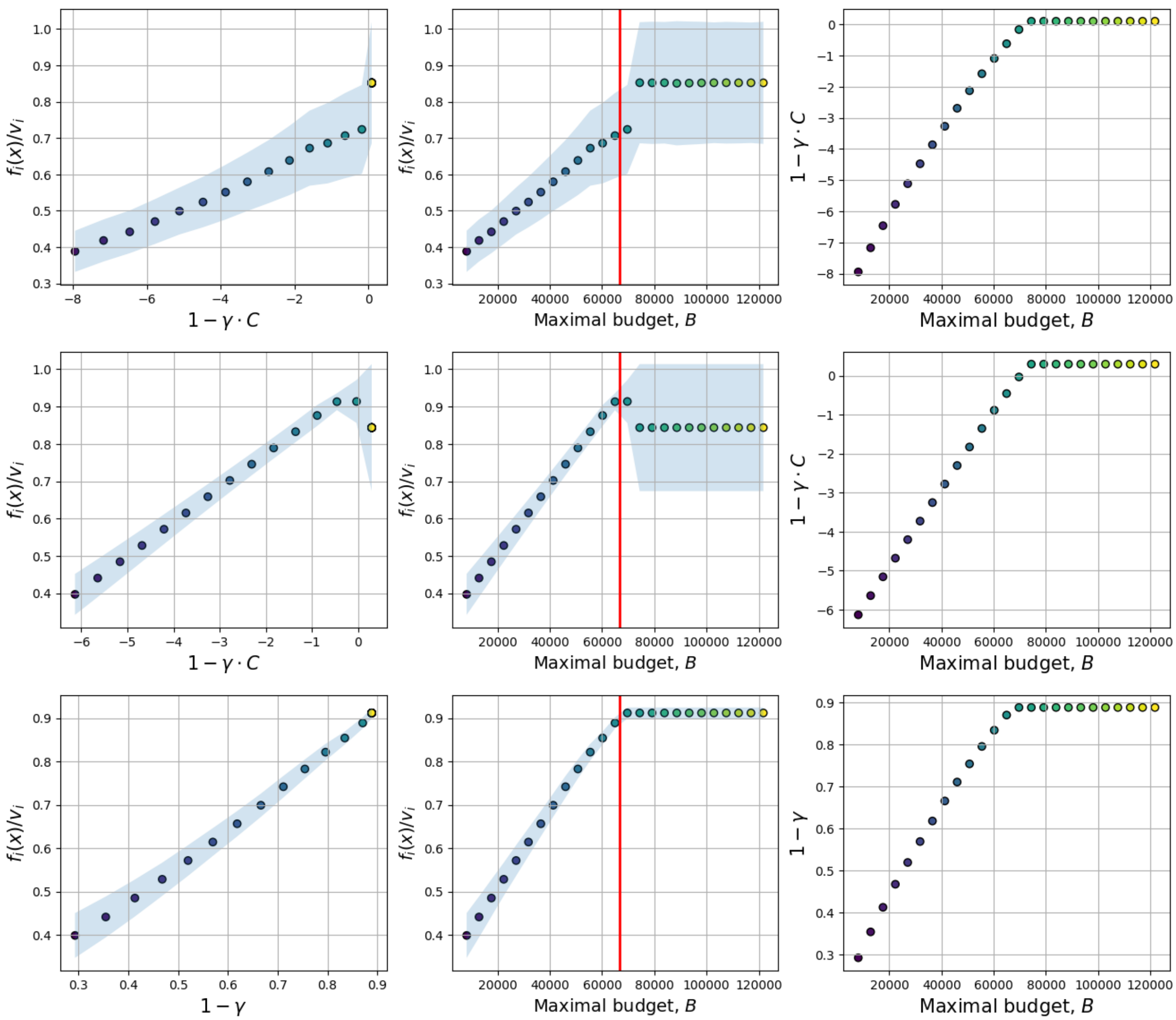

Рис. 2: The dependence of the function relative increments on the budget and the resulting γ for scenarios with differet budget. Upper is our solution, lower is exact solution.

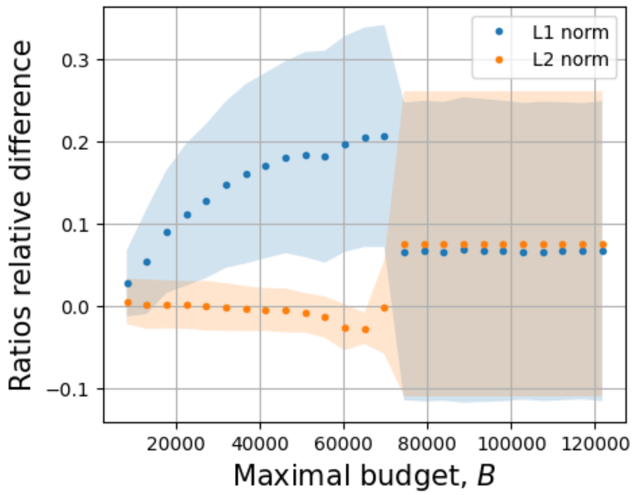

Рис. 3: Comparison of the result of the algorithm with respect to the exact solution. We see that the L2 solution is not much inferior.

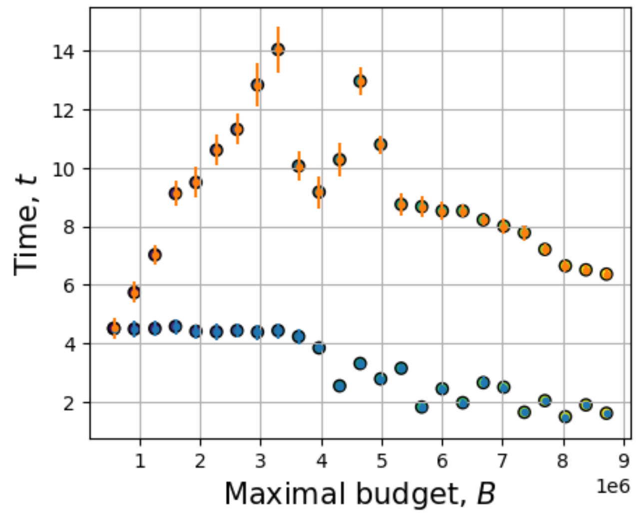

Рис. 4: Comparing the working time on different L2 budgets of an approximate solution with an accurate one. We see that the approximate solution is faster.

where N is ... L_D is ...

допи как будет известно что делать с постановкой задачи

in this formulation it can be infeasible due to constraint with b. To make it feasible for all bandwidths b we reform problem:

$$\min_F c^T F\mathbf{1} + c_a^T y \tag{MCF}$$
$$\text{s.t. } F\mathbf{1} \le b + y$$
$$NF = -L_D$$

This can be interpreted as additional bandwidth rent. We denote $g(b, D) = \text{value}(MCF(b, D))$.

proo

We are ready to introduce problem of this case. There some company, that provides package delivery service in some network. To work it needs to rent bandwidth. It can rent bandwidth at the beginnig of period with cost c_b and rent additional bandwidth within period for cost c_a. $c_b \le c_a$ componentwise in practice, because it is rented in advance for a long time. In every time period there are demands D_i, that set the scenario for this peroid. Of course company don't know this correspondence matrices. For m periods it rent some b_i bandwidth and observe D_i. At the end of period it observe its spendings $f_i(b_i) = c_b^T b_i + g(b_i, D_i)$. This functions is satisfy Lipshitz conditions. Company can deside to arrend bandwidth b at the beginnig of the perion with budget constraint $c_b^T b \le B$ so that it does not significantly change spending on already known scenarios. We get the T_1.

5.3 Different metrics

In this subsection we add another metric – maximal flow into considered metrics. Consider a network with m nodes and n edges. With each edge (i, j) we accociate upper bound of flow b_{ij}. In such network we wish to find the maximal amount of flow, which can be provided from node i to node j. This problem can be represented as follows from chapter 12 of [3]:

$$\begin{aligned} &\max_f \\ &\text{s.t. } 0 \le x \le b \\ &\qquad Ax = f(e_i - e_j) \end{aligned} \tag{MF}$$

where A is node-arc incidence matrix.

Such a metric may be needed to ensure the stability of the network under high loads.

6 Discussion

7

Список литературы

[1] Fulya Altiparmak, Mitsuo Gen, Lin Lin, and Turan Paksoy. A genetic algorithm approach for multi-objective optimization of supply chain networks. *Computers & industrial engineering*, 51(1):196–215, 2006.

[2] JC Moreno Banos and Markos Papageorgiou. A linear programming approach to large-scale linear optimal control problems. *IEEE transactions on automatic control*, 40(5):971–977, 1995.

[3] Mokhtar S Bazaraa, John J Jarvis, and Hanif D Sherali. *Linear programming and network flows*. John Wiley & Sons, 2011.

[4] Emil Bjornson, Eduard Axel Jorswieck, Mérouane Debbah, and Bjorn Ottersten. Multiobjective signal processing optimization: The way to balance conflicting metrics in 5g systems. *IEEE Signal Processing Magazine*, 31(6):14–23, 2014.

[5] Yadira Boada, Gilberto Reynoso-Meza, Jesús Picó, and Alejandro Vignoni. Multi-objective optimization framework to obtain model-based guidelines for tuning biological synthetic devices: an adaptive network case. *BMC systems biology*, 10:1–19, 2016.

[6] Samuel Daulton, David Eriksson, Maximilian Balandat, and Eytan Bakshy. Multi-objective bayesian optimization over high-dimensional search spaces. In *Uncertainty in Artificial Intelligence*, pages 507–517. PMLR, 2022.

[7] Mohammed Elmusrati, Hassan El-Sallabi, and Heikki Koivo. Applications of multi-objective optimization techniques in radio resource scheduling of cellular communication systems. *IEEE Transactions on Wireless Communications*, 7(1):343–353, 2008.

[8] Bennet Gebken and Sebastian Peitz. An efficient descent method for locally lipschitz multiobjective optimization problems. *Journal of Optimization Theory and Applications*, 188(3):696–723, 2021.

[9] F Gembicki and Y Haimes. Approach to performance and sensitivity multiobjective optimization: The goal attainment method. *IEEE Transactions on Automatic control*, 20(6):769–771, 1975.

[10] S. Greco. *Multiple Criteria Decision Analysis: State of the Art Surveys*. International Series in Operations Research & Management Science. Springer New York, 2006.

[11] M Murat Köksalan, Jyrki Wallenius, and Stanley Zionts. *Multiple criteria decision making: from early history to the 21st century*. World Scientific, 2011.

[12] Abdullah Konak, David W Coit, and Alice E Smith. Multi-objective optimization using genetic algorithms: A tutorial. *Reliability engineering & system safety*, 91(9):992–1007, 2006.

[13] Olvi L Mangasarian and T-H Shiau. Lipschitz continuity of solutions of linear inequalities, programs and complementarity problems. *SIAM Journal on Control and Optimization*, 25(3):583–595, 1987.

[14] R Timothy Marler and Jasbir S Arora. Survey of multi-objective optimization methods for engineering. *Structural and multidisciplinary optimization*, 26:369–395, 2004.

[15] Richard Kipp Martin. *Large scale linear and integer optimization: a unified approach*. Springer Science & Business Media, 2012.

[16] Ernesto Mastrocinque, Baris Yuce, Alfredo Lambiase, and Michael S Packianather. A multi-objective optimization for supply chain network using the bees algorithm. *International Journal of Engineering Business Management*, 5:38, 2013.

[17] Kaisa Miettinen. *Nonlinear multiobjective optimization*, volume 12. Springer Science & Business Media, 1999.

[18] Kaisa Miettinen, Francisco Ruiz, and Andrzej P Wierzbicki. Introduction to multiobjective optimization: interactive approaches. In *Multiobjective optimization: interactive and evolutionary approaches*, pages 27–57. Springer, 2008.

[19] Patrick Ngatchou, Anahita Zarei, and A El-Sharkawi. Pareto multi objective optimization. In *Proceedings of the 13th international conference on, intelligent systems application to power systems*, pages 84–91. IEEE, 2005.

[20] Gade Pandu Rangaiah and AB Petriciolet. Multi-objective optimization in chemical engineering. *Developments and applications/edited by Gade Pandu Rangaiah, Adrián Bonilla-Petriciolet*, 2013.

[21] Ozan Sener and Vladlen Koltun. Multi-task learning as multi-objective optimization. *Advances in neural information processing systems*, 31, 2018.

[22] Thorsten Suttorp and Christian Igel. Multi-objective optimization of support vector machines. *Multi-objective machine learning*, pages 199–220, 2006.

[23] Shinya Suzuki, Shion Takeno, Tomoyuki Tamura, Kazuki Shitara, and Masayuki Karasuyama. Multi-objective bayesian optimization using pareto-frontier entropy. In *International conference on machine learning*, pages 9279–9288. PMLR, 2020.

[24] Marcela Zuluaga, Guillaume Sergent, Andreas Krause, and Markus Püschel. Active learning for multi-objective optimization. In *International conference on machine learning*, pages 462–470. PMLR, 2013.