How to advance in Structural Convex Optimization

Yurii Nesterov

October, 2008

Abstract

In this paper we are trying to analyze the common features of the recent advances in Structural Convex Optimization: polynomial-time interior-point methods, smoothing technique, minimization in relative scale, and minimization of composite functions.

Keywords convex optimization · non-smooth optimization · complexity theory · black-box model · optimal methods · structural optimization · smoothing technique.

Mathematics Subject Classification (2000) 90C06 · 90C22 · 90C25 · 90C60.

Convex Optimization is one of the rare fields of Numerical Analysis, which benefit from existence of well-developed complexity theories. In our domain, this theory was created in the middle of the seventies in a series of papers by A.Nemirovsky and D.Yudin (see [8] for full exposition). It consists of three parts:

– Classification and description of problem instances.
– Lower complexity bounds.
– Optimal methods.

In [8], the complexity of a convex optimization problem was linked with its *level of smoothness* introduced by Hölder conditions on the first derivatives of functional components. It was assumed that the only information the optimization methods can learn about the particular problem instance is the values and derivatives of these components at some test points. This data can be reported by a special unit called *oracle*, and it is *local*, which means that it is not changing if the function is modified far enough from the test point. This model of interaction between the optimization scheme and the problem data is called the *local Black Box*. At the time of its development, this concept fitted very well the existing computational practice, where the interface between the general optimization packages and the problem data was established by Fortran subroutines created independently by the users.

Black-Box framework allows to speak about the lower performance bounds for diffierent problem classes in terms of *informational complexity*. That is the lower estimate for the number of calls of oracle which is necessary for any optimization method in order to guarantee delivering an ε-solution to any problem from the problem class. In this performance measure we do not include at all the complexity of auxiliary computations of the scheme.

Let us present these bounds for the most important classes of optimization problems posed in the form

$$\min_{x \in Q} f(x), \qquad (1)$$

where $Q \subseteq R^n$ is a bounded closed convex set ($||x|| \leq R$, $x \in Q$), and function f is convex on Q. In the table below, the first column indicates the problem class, the second one gives an upper bound for allowed number of calls of the oracle in the optimization scheme[1], and the last column gives the lower bound for analytical complexity of the problem class, which depends on the absolute accuracy ε and the class parameters.

This paper was written during the visit of the author at IFOR (ETH, Zurich). The author expresses his gratitude to the Scientific Director of this center Hans-Jacob Lüthi for his support and excellent working conditions.

Yurii Nesterov
Center for Operations Research and Econometrics (CORE)
Université catholique de Louvain (UCL)
34 voie du Roman Pays, 1348 Louvain-la-Neuve, Belgium.
Tel.: +32-10-474348
Fax: +32-10-474301
e-mail: Yurii.Nesterov@uclouvain.be

Problem class	Limit for calls	Lower bound
$\mathcal{C}_1: \|\nabla f(\cdot)\| \le L$	$\le O(n)$	$O\left(\frac{L^2R^2}{\epsilon^2}\right)$
$\mathcal{C}_2: \|\nabla^2 f(\cdot)\| \le M$	$\le O(n)$	$O\left(\frac{M^{1/2}R}{\epsilon^{1/2}}\right)$
$\mathcal{C}_3: \|\nabla f(\cdot)\| \le L$	$\ge O(n)$	$O\left(n \ln \frac{LR}{\epsilon}\right)$

(2)

It is important that these bounds are *exact*. This means that there exist methods, which have efficiency estimates on corresponding problem classes proportional to the lower bounds. The corresponding *optimal methods* were developed in [8,9,19,22,23]. For further references, we present a simplified version of the optimal method [9] as applied to the problem (1) with $f \in C_2$:[2]

Choose a starting point $y_0 \in Q$ *and set* $x_{-1} = y_0$. *For* $k \ge 0$ *iterate:*

$$x_k = \arg\min_{x \in Q} \left[f(y_k) + \langle \nabla f(y_k), x - y_k \rangle + \frac{M}{2}\|x - y_k\|^2 \right],$$
$$y_{k+1} = x_k + \frac{k}{k+3}(x_k - x_{k-1}). \quad (3)$$

As we see, the complexity of each iteration of this scheme is comparable with that of the simplest gradient method. However, the rate of convergence of method (3) is much faster.

After a certain period of time, it became clear that, despite its mathematical excellence, Complexity Theory of Convex Optimization has a hidden drawback. Indeed, in order to apply convex optimization methods, we need to be sure that functional components of our problem are convex. However, we can check convexity only by analyzing the *structure* of these functions:[3] If our function is obtained from the *basic* convex functions by *convex* operations (summation, maximum, etc.), we conclude that it is convex. If not, then we have to apply general optimization methods which usually do not have theoretical guarantees for the global performance.

Thus, the functional components of the problem are not in the black box the moment we check their convexity and choose minimization scheme. However, we put them into the black box for numerical methods. That is the main conceptual contradiction of the standard Convex Optimization.

Intuitively, we always hope that the structure of the problem can be used for improving the performance of minimization schemes. Unfortunately, structure is a very fuzzy notion, which is quite difficult to formalize. One possible way to describe the structure is to fix the *analytical type* of functional components. For example, we can consider the problems with linear constraints only. It can help, but this approach is very fragile: If we add just a single constraint of another type, then we get a new problem class, and all theory must be redone from scratch.

On the other hand, it is clear that having the structure at hand we can play a lot with the *analytical form* of the problem.We can rewrite the problem in many equivalent settings using non-trivial transformations of variables or constraints, introducing additional variables, etc. However, this would serve almost no purpose without fixing a clear final goal. So, let us try to understand what it could be.

As usual, it is better to look at classical examples. In many situations the sequential reformulations of the initial problem can be seen as a part of numerical scheme. We start from a complicated problem $\mathcal{P}$ and, step by step, change its structure towards to the moment we get a trivial problem (or, a problem which we know how to solve):

$$\mathcal{P} \to \ldots \to (f^*, x^*).$$

A good example of such a strategy is the standard approach for solving system of linear equations

$$Ax = b.$$

We can proceed as follows:

1. Check if A is symmetric and positive definite. Sometimes this is clear from the origin of the matrix.
2. Compute Cholesky factorization of this matrix:
   $A = LL^T$;
   where L is a lower-triangular matrix.
   Form two auxiliary systems
   $Ly = b$, $L^T x = y$.
3. Solve these system by sequential exclusion of variables.

Imagine for a moment that we do not know how to solve the system of linear equations. In order to discover the above scheme we should apply the following

Golden Rules
1. Find a class of problems which can be solved very efficiently.[a] 2. Describe the transformation rules for converting the initial problem into desired form. 3. Describe the class of problems for which these transformation rules are applicable.
[a] In our example, it is the class of linear systems with triangular matrices.

(4)

In Convex Optimization, these rules were used already several times for breaking down the limitations of Complexity Theory.

Historically, the first example of that type is the theory of polynomial-time interior-point methods (IPM) based on *self-concordant barriers*. In this framework, the class of easy problems is formed by problems of unconstrained minimization of self-concordant functions treated by the Newton method. This *know-how* is further used in the framework of path-following schemes for solving so-called *standard* minimization problems. Finally, it can be shown that by a simple barrier calculus this approach can be extended onto all convex optimization problems with known structure (see [11,18] for details). The efficiency estimates of corresponding schemes are of the order $O(\upsilon^{1/2} \ln \frac{\upsilon}{\epsilon})$ iterations of the Newton method, where υ is the parameter of corresponding self-concordant barrier. Note that for many important feasible sets this parameter is smaller than the dimension of the space of variables. Hence, for the pure Black-Box schemes such an efficiency is simply unreachable in view of the lower complexity bound for class C_3 (see (2)). It is interesting that formally the modern IPMs look very similar to the usual Black-Box schemes (Newton method plus path-following approach), which were developed in the very early days of Nonlinear Optimization [4]. However, this is just an illusion. For complexity analysis of *polynomial-time* IPM, it is crucial that

[1] **If this upper bound is smaller than $O(n)$, then the dimension of the problem is really very big, and we cannot afford the method to perform this amount of calls.**

[2] **In method (11)-(13) from [9], we can set $a_k = 1 + k/2$ since in the proof we need only to ensure $a_k^2 + 1 - a_k^2 \le a_k + 1$.**

[3] **Numerical verification of convexity is an extremely difficult problem.**

they employ the special barrier functions which *do not satisfy* the local Black-Box assumptions (see [10] for discussion).

The second example of using the rules (4) needs more explanations. By certain circumstances, these results were discovered with a delay of twenty years. Perhaps they were too simple. Or maybe they are in a seemingly very sharp contradiction with the rigorously proved lower bounds of Complexity Theory.

Anyway, now everything looks almost evident. Indeed, in accordance to Rule 1 in (4), we need to find a class of very easy problems. And this class can be discovered *directly* in Table (2)! To see that, let us compare the complexity of the classes $C1$ and $C2$ for the accuracy of 1% ($\varepsilon = 10^{-2}$). Note that in this case, the accuracy-dependent factors in the efficiency estimates vary from ten to ten thousands. So, the natural question is:

Can the easy problems *from* C_2 *help us somehow in finding an approximate solution to the* difficult problems *from* C_1?

And the evident answer is: Yes, of course! It is a simple exercise in Calculus to show that we can always approximate a Lipschitz-continuous nonsmooth convex function on a bounded convex set with a uniform accuracy $\varepsilon > 0$ by a smooth convex function with Lipschitz-continuous gradient. We pay for the accuracy of approximation by a large Lipschitz constant M for the gradient, which should be of the order $O(\frac{1}{\epsilon})$. Putting this bound for M in the efficiency estimate of C_2 in (2), we can see that in principle, it is possible to minimize nonsmooth convex functions by the oracle-based gradient methods with analytical complexity $O(\frac{1}{\epsilon})$. But what about the Complexity Theory? It seems that it was *proved* that such efficiency is just impossible.

It is interesting that in fact we do not get any contradiction. Indeed, in order to minimize a smooth approximation of nonsmooth function by an oracle-based scheme, we need to change the initial oracle. Therefore, from mathematical point of view, we violate the Black-Box assumption. On the other hand, in the majority of practical applications this change is not difficult. Usually we can work directly with the structure of our problem, at least in the cases when it is created by us.

Thus, the basis of the *smoothing technique* [12,13] is formed by two ingredients: the above observation, and a trivial but systematic way for approximating a nonsmooth function by a smooth one. This can be done for convex functions represented explicitly in a max-form:

$$f(x) = \max_{u \in Q_d} \{ \langle Ax - b, u \rangle - \phi(u),$$

where Q_d is a bounded and convex dual feasible set and $\phi(u)$ is a concave function. Then, choosing a nonnegative strongly convex function $d(u)$, we can define a smooth function

$$f_\mu(x) = \max_{u \in Q_d} \{ \langle Ax - b, u \rangle - \phi(u) - \mu \cdot d(u) \} \quad (5)$$

which approximates the initial objective. Indeed, denoting $D_d = \max_{u \in Q_d} d(u)$, we get

$$f(x) \geq f_\mu(x) \geq f(x) \ \ \mu D_d.$$

At the same time, the gradient of function f_μ is Lipschitz-continuous with Lipschitz constant of the order of $O(\frac{1}{\mu})$ (see [12]) for details).

Thus, we can see that for an *implementable* definition (5), we get a possibility to solve problem (1) in $O(\frac{1}{\epsilon})$ iterations of the fast gradient method (3). In order to see the magnitude of the improvement, let us look at the following example:

$$\min_{x \in \Delta_n} \left[f(x) \stackrel{\text{def}}{=} \max_{1 \leq j \leq m} \langle a_j, x \rangle \right], \quad (6)$$

where $\Delta_n \in R^n$ is a standard simplex. Then the properly implemented smoothing technique ensures the following rate of convergence:

$$f(x_N) - f^* \leq \frac{4\sqrt{\ln n \cdot \ln m}}{N} \cdot \max_{i,j} |a_j^{(i)}|.$$

If we apply to problem (6) the standard subgradient methods (e.g. [14]), we can guarantee only

$$f(x_N) - f^* \leq \frac{\sqrt{\ln n}}{\sqrt{N+1}} \cdot \max_{i,j} |a_j^{(i)}|.$$

Thus, up to a logarithmic factor, for obtaining the same accuracy, the methods based on smoothing technique need only a square root of iterations of the usual subgradient scheme. Taking into account, that usually the subgradient methods are allowed to run many thousands or even millions of iterations, the gain of the smoothing technique in computational time can be enormously big.[4]

It is interesting, that for problem (6) the computation of the smooth approximation is very cheap. Indeed, let us use for smoothing the *entropy function*:

$$d(u) = \ln m + \sum_{i=1}^{n} u^{(i)} \ln u^{(i)}, \quad u \in \Delta_m.$$

Then the smooth approximation (5) of the objective function in (6) has the following compact representation:

$$f_\mu(x) = \mu \ln \left[\frac{1}{m} \sum_{j=1}^{m} e^{\langle a_j, x \rangle / \mu} \right].$$

Thus, the complexity of the oracle for $f(x)$ and $f_\mu(x)$ is similar. Note that again, as in the polynomial-time IPM theory, we apply the standard oracle-based method ((3) in this case) to a function which does not satisfy the Black-Box assumptions.

An inexplicable blindness to the possibility to reduce the complexity of nonsmooth optimization problems with known structure is not restricted to the smoothing technique only. As it was shown in [7], very similar results can be obtained by the extra-gradient method by G. Korpelevich [6] using the fact that this method is optimal for the class of variational inequalities with Lipschitz-continuous operator (for these problems it converges as $O(\frac{1}{k})$). Actually, in a verbal form, the optimality of the extra-gradient method was known already for a couple of decades. However, a rigorous proof of this important fact and discussion of its consequences for Structural Nonsmooth Optimization was published only in [7], after discovering the smoothing technique.

To conclude this section, let us discuss the last example of acceleration strategies in Structural Optimization. Consider the problem of minimizing the *composite*

[4] **It is easy to see that the standard subgradient methods for nonsmooth convex minimization need indeed $O(\frac{1}{\epsilon^2})$ operations to converge. Consider a univariate function $f(x) = |x|$, $x \in R$. Let us look at the subgradient process:**

$$x_{k+1} = x_k - h_k f'(x_k), \quad x_0 = 1, \quad h_k = \frac{1}{\sqrt{k+1}} + \frac{1}{\sqrt{k+2}}, \quad k \geq 0.$$

It easy to see that $|x_k| = \frac{1}{\sqrt{k+1}}$. However, the step-size sequence is optimal [8].

objective function:

$$\min_{x \in R^n} [\, f(x) + \Psi(x) \,], \qquad (7)$$

where the function f is a convex differentiable function on dom Ψ with Lipschitz-continuous gradient, and function Ψ is an *arbitrary* closed convex function. Since Ψ can be even discontinuous, in general this problem is very difficult. However, if we assume that function Ψ is *simple*, then the situation is changing. Indeed, suppose that for any $\bar{y} \in$ dom Ψ we are able to solve explicitly the following auxiliary optimization problem:

$$\min_{x \in \mathrm{dom}\, \Psi} \left[f(\bar{y}) + \langle \nabla f(\bar{y}), x - \bar{y} \rangle + \tfrac{M}{2} \|x - \bar{y}\|^2 + \Psi(x) \right] \qquad (8)$$

(compare with (3)). Then it becomes possible to develop for problem (7) fast gradient methods (similar to (3)), which have the rate of convergence of the order $O(\frac{1}{k^2})$ (see [15] for details; similar technique was developed in [3]). Note that the formulation (7) can be also seen as a part of Structural Optimization since we use the knowledge of the structure of its objective function directly in the optimization methods.

Conclusion

In this paper, we have considered several examples of significant acceleration of the usual oracle-based methods. Note that the achieved progress is visible only because of the supporting complexity analysis. It is interesting that all these methods have some prototypes proposed much earlier:

– Optimal method (3) is very similar to the *heavy point* method:

$$x_{k+1} = x_k - \alpha \nabla f(x_k) + \beta(x_k - x_{k-1}),$$

where α and β are some fixed positive coefficients (see [20] for historical details).

– Polynomial-time IPM are very similar to *some* variants of the classical barrier methods [4].

– The idea to apply smoothing for solving minimax problems is also not new (see [21] and the references therein).

At certain moments of time, these ideas were quite new and attractive. However, they did not result in a significant change in computational practice since they were not provided with a convincing complexity analysis. Indeed, many other schemes have similar theoretical justifications and it was not clear at all why these particular suggestions deserve more attention. Moreover, even now, when we know that the modified variants of some old methods give excellent complexity results, we cannot say too much about the theoretical efficiency of the original schemes.

Thus, we have seen that in Convex Optimization the complexity analysis plays an important role in selecting the promising optimization methods among hundreds of others. Of course, it is based on investigation of the worst-case situation. However, even this limited help is important for choosing the perspective directions for further research. This is true especially now, when the development of Structural Optimization makes the problem settings and corresponding efficiency estimates more and more interesting and diverse.

The size of this paper does not allow us to discuss other interesting setting of Structural Convex Optimization (e.g. optimization in relative scale [16, 17]). However, we hope that even the presented examples can help the reader to find new and interesting research directions in this promising field (see, for example, [1,2,5]).

References

1. A. d'Aspremont, O. Banerjee, and L. El Ghaoui. First-Order Methods for Sparse Covariance Selection. *SIAM Journal on Matrix Analysis and its Applications*, 30(1), 56-66, (2008).
2. O. Banerjee, L. El Ghaoui, and A. d'Aspremont. Model Selection Through Sparse Maximum Likelihood Estimation. *Journal of Machine Learning Research*, 9, 485-516 (2008).
3. A. Beck and M. Teboulle. A Fast Iterative Shrinkage-Threshold Algorithm Linear Inverse Problems. Research Report, Technion (2008).
4. A.V. Fiacco and G.P. McCormick. Nonlinear Programming: Sequential Uncon strained Minimization Technique. *John Wiley*, New York, 1968.
5. S. Hoda, A. Gilpin, and J. Pena. Smoothing techniques for computing Nash equilibria of sequential games. Research Report. Carnegie Mellon University, (2008).
6. G.M. Korpelevich. The extragradient method for finding saddle points and other problems. *Matecon* 12, 747-756 (1976).
7. A. Nemirovski. Prox-method with rate of convergence $O(1/t)$ for variational inequalities with Lipschitz continuous monotone operators and smooth convex concave saddle point problems. *SIAM Journal on Optimization*, 15, 229-251 (2004).
8. A. Nemirovsky and D. Yudin. Problem Complexity and Method Efficiency in Optimization. *Wiley*, New-York, 1983.
9. Yu. Nesterov. A method for unconstrained convex minimization problem with the rate of convergence $O(\frac{1}{k^2})$. *Doklady AN SSSR* (translated as Soviet Mathematics Doklady), 269(3), 543-547 (1983).
10. Yu. Nesterov. Interior-point methods: An old and new approach to nonlinear progamming. *Mathematical Programming*, 79(1-3), 285-297 (1997).
11. Yu. Nesterov. Introductory Lectures on Convex Optimization. *Kluwer*, Boston, 2004.
12. Yu. Nesterov. Smooth minimization of non-smooth functions. CORE Discussion Paper 2003/12 (2003). Published in *Mathematical Programming*, 103 (1), 127-152 (2005).
13. Yu. Nesterov. Excessive gap technique in nonsmooth convex minimization. *SIAM Journal on Optimization,* 16 (1), 235-249 (2005).
14. Yu. Nesterov. Primal-dual subgradient methods for convex problems. *Mathematical Programming* (2007)
15. Yu. Nesterov. Gradient methods for minimizing composite objective function. *CORE Discussion Paper* 2007/76, (2007).
16. Yu. Nesterov. Rounding of convex sets and efficient gradient methods for linear programming problems. *Optimization Methods and Software*, 23(1), 109-135 (2008).
17. Yu. Nesterov. Unconstrained convex minimization in relative scale. Accepted by *Mathematics of Operations Research*.
18. Yu. Nesterov, A. Nemirovskii. Interior point polynomial methods in convex programming: Theory and Applications. *SIAM*, Philadelphia, 1994.
19. B.T. Polyak. A general method of solving extremum problems. *Soviet Mat. Dokl.* 8, 593-597 (1967)
20. B. Polyak. Introduction to Optimization. *Optimization Software*, New York, 1987.
21. R. Polyak. Smooth Optimization Methods for Minimax Problems. S*IAM J. Control and Optimization*, 26(6), 1274-1286 (1988).
22. N.Z. Shor. Minimization Methods for Nondifferentiable Functions. *Springer-Verlag*, Berlin, 1985.
23. S.P. Tarasov, L.G. Khachiyan, and I.I. Erlikh. The Method of Inscribed Ellipsoids. *Soviet Mathematics Doklady*, 37, 226– 230 (1988).