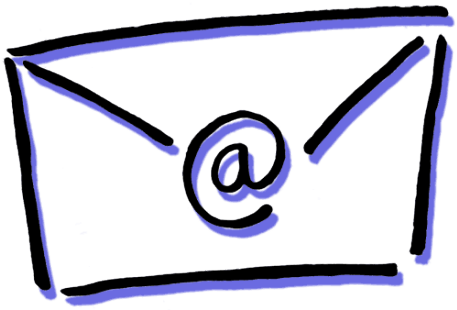
SIAG on Optimization Views and News

A Forum for the SIAM Activity Group on Optimization

Volume 34 Number 1 July 2026

Contents

Are you receiving this by postal mail?
Do you prefer electronic delivery?



Email siagoptnews@lists.mcs.anl.gov

Comments from the Editors

Dear SIAM Activity Group on Optimization,

We hope you were able to attend the recent SIAM Conference on Optimization. The meeting was an excellent convening of our community that hopefully fostered new and exciting directions for research. We also note that the occurrence of the conference generally aligns with a transition in the leadership of our SIAG/OPT and we welcome our new officers. Michael Ulbrich now serves as the SIAG/OPT's new Chair, and he has penned the Column Chair for this issue of Views and News.

As a research spotlight in this issue, we highlight work by Benjamin Grimmer, Kevin Shu and Alex L. Wang. The authors investigate a dynamic form of worst-case complexity that attempts to bound the worst case performance of an algorithm over only the set of problem instances consistent with first-order oracle queries already performed over the course of running the algorithm. Using a game-theoretic lens, they describe *subgame perfect* optimization methods that re-optimize their convergence proofs at runtime, leading to dynamically improved algorithms for smooth and nonsmooth convex problems. Their results are an intriguing bridge between classical minimax theory, performance estimation, and practical adaptivity.

All issues of *Views and News* are available online at https://siagoptimization.github.io/ViewsAndNews.

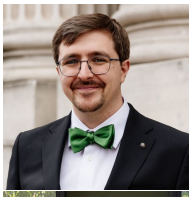
The SIAG on Optimization Views and News mailing list, where editors can be reached for feedback, is siagoptnews@lists.mcs.anl.gov. Suggestions for bulletin notes of interest to SIAG/OPT members, as well as research highlight articles are always welcome.

Jan Kronqvist
KTH Royal Institute of Technology
Email: jankr@kth.se

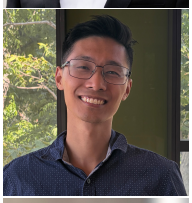
Matt Menickelly
Argonne National Laboratory
Email: mmenickelly@anl.gov

Articles

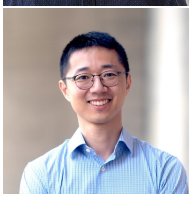
Subgame Perfection and Dynamic Optimality in Optimization



Benjamin Grimmer
Department of Applied Mathematics and Statistics
Johns Hopkins University
grimmer@jhu.edu
https://www.ams.jhu.edu/~grimmer/



Kevin Shu
Department of Combinatorics and Optimization
University of Waterloo
k6shu@waterloo.ca
https://kevinshu.me/



Alex L. Wang
Daniels School of Business
Purdue University
wang5984@purdue.edu
https://www.a-l-wang.com/

1 Introduction

Traditional optimization algorithm design has focused on minimax optimal algorithms—that is, those with the best worst-case performance. While worst-case guarantees are conceptually important, problems in practice are rarely adversarial. For this reason, we would like algorithms that both retain worst-case performance guarantees and exploit non-adversarial problem features. In short, we want to be 'as performant as possible, given the input'.

However, this is difficult to formalize in a satisfying manner. Online algorithms are often assessed by regret [24]. Regret compares an algorithm's aggregate performance when encountering problem data online against the single best solution given all data had been made available up front. Alternative notions of instance optimality have also been developed in the computer science literature, e.g., [1]. None of these map perfectly onto the traditional oracle access model of optimization.

Here, we present a game-theoretic approach, first developed in [10], rooted in the classical oracle access model. Taking this approach, we guarantee being 'as performant as possible, given the first-order information seen so far'.

The information-theoretic oracle model, developed in the seminal work of Nemirovski and Yudin [21], allows us to ask what the best algorithm from a class of algorithms can do against any problem from a class of problems. For example, considering gradient methods as the class of algorithms and smooth convex minimization as the class of problems, one might ask: Given a fixed budget of gradient queries, how small can one make the final objective gap or gradient norm?

Questions like in the above example take an *a priori* view. By this, we mean they aim to fix the algorithm in advance of the problem instance and interrogate the algorithm's performance against the worst problem in the class. Such studies have led to insights like the discovery of Nesterov's famous order-optimal momentum method [22].

Simultaneously, the information-theoretic oracle model enables a much less studied *dynamic* question. Supposing an algorithm has run for n iterations out of a budget of $N > n$, what is the worst case against all remaining possible problem instances (i.e., only those consistent with the n first-order queries already revealed)? The related dynamic algorithm design question is then: Given n oracle responses from the first n iterations, what is the best possible algorithm to employ for the remaining $N - n$ iterations?

When $n = 0$, this question asks about the classical "a priori" notion of minimax optimal algorithm design. For $n > 0$, this is a dynamic question to be answered at runtime in response to problem data. This article discusses recent advances showing that this notion of dynamic optimality is tractably achievable in certain settings.

2 The Classic Oracle Model

First, we formalize the information-theoretic oracle model for optimization. Throughout, we restrict to first-order methods for

$$f_\star = \min_{x\in\mathbb{R}^d} f(x) \tag{1}$$

with $f\colon \mathbb{R}^d \to \mathbb{R}\cup\{\infty\}$, attained at some $x_\star$.

2.1 Problem and Algorithm Classes

To fix an algorithm design problem, we require a fixed family of functions f and a fixed family of algorithms interacting with f via an oracle.

Problem Classes We denote a family of problem instances by $\mathtt{P}$, where $\mathtt{p} \in \mathtt{P}$ denotes a particular instance. For example, we denote the L-smooth convex optimization problems by

$$\begin{aligned}\mathtt{P}_L = \{(f,x_0)\ |&f \text{ is convex, has } L\text{-Lipschitz}\\ &\text{gradient, and attains its}\\ &\text{minimum at some } x_\star\}\end{aligned} \tag{2}$$

with each problem specified by its objective function and an initialization $x_0 \in \mathbb{R}^d$. As a second example from nonsmooth optimization, one may consider the family of Lipschitz functions with bounded initial distance to a minimizer

$$\begin{aligned}\mathtt{P}_{M,D} = \{(f,x_0)\ |&f \text{ is convex, } M\text{-Lipschitz, and}\\ &\text{attains its minimum at some}\\ &x_\star \text{ with } \|x_0 - x_\star\|_2 \le D\}.\end{aligned} \tag{3}$$

Note that in the above problem classes, we have not fixed the dimension d. As a result, these families contain problem instances in every dimension d.

Algorithm Classes We consider algorithms given access to an oracle providing information about the problem instance $\mathtt{p} \in \mathtt{P}$. An N-step algorithm provides a rule for iteratively constructing the sequence of query points x_n with $n = 0, 1, \ldots, N$ based on the oracle's responses to all past queries.

As a first example, consider gradient methods for smooth convex problems (2). The class of all deterministic N-step first-order methods, generating x_n as a function of oracle responses $\{(x_i, f_i, g_i)\}_{i=0}^{n-1}$ where $f_i = f(x_i)$ and $g_i = \nabla f(x_i)$, is denoted $\mathtt{A}_{\mathtt{det}}(N)$. Another commonly studied class is the class of gradient span methods, restricted to selecting

$$x_n \in x_0 + \operatorname{span}\{g_0, \ldots, g_{n-1}\}, \tag{4}$$

denoted $\mathtt{A}_{\mathtt{span}}(N)$. As a subset of either algorithm class, one may consider "fixed-step" first-order methods with predetermined span coefficients, recursively setting

$$x_n = x_{n-1} - \sum_{i=0}^{n-1} H_{n,i} g_i, \tag{5}$$

denoted $\mathtt{A}_{\mathtt{fixed}}$. This last family is computationally convenient as each algorithm can be represented by an $N \times N$ lower triangular H matrix.

As examples in nonsmooth optimization, one commonly considers the replacement of gradient oracles with either subgradient queries or evaluations of a proximal operator.

2.2 Minimax Optimality

Families of algorithms and problems are the central objects of the study of algorithm design and analysis. In particular, minimax optimal algorithms and problems play dual roles to each other. Below, we make this dual relationship mathematically precise.

Given problem and algorithm classes, $\mathtt{P}$ and $\mathtt{A}$, one can ask for an algorithm with the best worst-case, i.e., attaining

$$\min_{\mathtt{a} \in \mathtt{A}} \max_{\mathtt{p} \in \mathtt{P}} M(\mathtt{a}, \mathtt{p}) \tag{6}$$

where $M(\mathtt{a}, \mathtt{p})$ is some measure of algorithm performance like the final objective gap at x_N when $\mathtt{a}$ is applied to $\mathtt{p}$. For the particular setting of $\mathtt{P}_L$ and $\mathtt{A}_{\mathtt{span}}(N)$, a commonly considered design problem optimizes the relative final gap

$$\min_{\mathtt{a} \in \mathtt{A}_{\mathtt{span}}(N)} \max_{\mathtt{p} \in \mathtt{P}_L} \frac{f(x_N) - f_\star}{\frac{L}{2}\|x_0 - x_\star\|_2^2}. \tag{7}$$

Note that since the class $\mathtt{P}_L$ imposes no limit on the placement of x_0, a relative measure is needed.

The dual optimization problem seeks a problem instance maximally hard for all algorithms within a class. That is,

$$\max_{\mathtt{p} \in \mathtt{P}} \min_{\mathtt{a} \in \mathtt{A}} M(\mathtt{a}, \mathtt{p}). \tag{8}$$

By weak duality, the problem of designing hard instances (8) provides lower bounds on the problem of designing an optimal algorithm (6).

Remarkably, strong duality holds between these min-max problems for many choices of $\mathtt{P}$, $\mathtt{A}$, and $M(\mathtt{a}, \mathtt{p})$.

2.3 A History of Minimax Optimality

For smooth convex optimization with $\mathtt{P}_L$ and $\mathtt{A}_{\mathtt{span}}(N)$, this strong duality was observed up to constants by Nesterov [22] and Nemirovski and Yudin [21] where algorithms and hard instances with matching $O(1/N^2)$ and $\Omega(1/N^2)$ rates were demonstrated. This strong duality was demonstrated exactly by the minimax optimal Optimized Gradient Method (OGM) of [16] and maximin hard instance of [5].

For nonsmooth optimization via subgradient methods, the classical subgradient method is precisely minimax optimal with an exactly matching maximin hard instance (see [7, Appendix A]). For proximal point and proximal gradient methods, strong duality holds as well, being attained by Güler's method [13] and the OptISTA method [14] and the hard instances in [14].

Among nonexpansive fixed-point methods (equivalently, monotone operators), exactly optimal algorithms were demonstrated by [18, 15] with a matching hard instance due to [25]. A complete characterization of the set of optimal fixed-step methods was given by [28].

In all of the above settings, the pursuit of minimax optimal algorithms led to new insights into how to design fast methods. The discovery of exactly optimal methods is fundamentally tied to the Performance Estimation Problem (PEP) framework [8, 27, 26]. PEPs in their primal form provide a direct way to evaluate the inner maximization of (6) via a semidefinite program for fixed-step methods (5). Perhaps more importantly, the dual PEP provides an optimal, structured convergence proof.

3 Dynamic Optimality

Each minimax optimal method discussed so far takes the fixed-step form (5). This form precludes any dynamic updates or adaptation at runtime. Such adjustments are not needed to optimize worst-case performance but are essential for practical performance. If an algorithm encountered a zero gradient, it ought to halt as it has found a minimizer. Alas, fixed-step momentum methods may get carried away.

As a failure case, consider minimizing $f(x) = \frac{1}{2}x^2$. Applied to this problem, OGM converges at its worst-case rate, while smarter methods will identify the minimizer in exactly two steps. See [10] for the details of this example.

3.1 Subgame Perfect Methods

The information-theoretic model provides us with language to describe dynamic optimality. Suppose an algorithm has run up to iteration $n \leq N$, generating first-order information

$$\mathcal{H}_n = ((x_i, f_i, g_i))_{i=0}^{n-1}.$$

One can consider the subclass of problems $\mathtt{P}^{\mathcal{H}_n} \subseteq \mathtt{P}$ restricted to have $f_i = f(x_i)$ and $g_i \in \partial f(x_i)$ for $i = 0, \ldots, n-1$. Likewise one can consider the subclass of algorithms $\mathtt{A}^{\mathcal{H}_n} \subseteq \mathtt{A}$ which reproduce $x_0, \ldots, x_{n-1}$ given the oracle responses in $\mathcal{H}_n$.

Then the design problem for how an algorithm should best proceed from iteration n, conditioned on the given observa-

tions $\mathcal{H}_n$, is

$$\min_{\mathtt{a}\in\mathtt{A}^{\mathcal{H}_n}} \max_{\mathtt{p}\in\mathtt{P}^{\mathcal{H}_n}} M(\mathtt{a},\mathtt{p}). \tag{9}$$

Definition 1. *We say that an algorithm* $\mathtt{a}\in\mathtt{A}$ *is subgame perfect against a problem class* $\mathtt{P}$ *at measure* M *if for any history* $\mathcal{H}_n$ *it generates,* $\mathtt{a}$ *attains the optimal value of* (9).

Subgame perfection is clearly stronger than minimax optimality. Given $n=0$, $\mathcal{H}_n$ contains no information and so attainment of (9) is exactly attainment of (6). Beyond this base case, subgame perfect methods must ensure they remain optimal against each observed problem subclass $\mathtt{P}^{\mathcal{H}_n}$ for $n>0$. Dually, one can consider the dynamic design of hard problem instances via

$$\max_{\mathtt{p}\in\mathtt{P}^{\mathcal{H}_n}} \min_{\mathtt{a}\in\mathtt{A}^{\mathcal{H}_n}} M(\mathtt{a},\mathtt{p}). \tag{10}$$

Just as strong duality between (6) and (8) was key historically to identifying minimax optimal algorithms, strong duality between the dynamic design problems (9) and (10) plays an analogous role in establishing subgame perfect methods.

3.2 The Minimization Game

This dynamic notion of optimality can be understood by considering minimization as an iterative zero-sum game. Consider an iterative game between Alice "the Algorithm" and Bob "the Bad instance". At round n of the game, Alice first declares a point x_n satisfying the rules of the given algorithm class $\mathtt{A}$ (e.g., lying in the span (4)). Bob then provides oracle evaluations at that point satisfying the consistency constraint that some $\mathtt{p}\in\mathtt{P}$ agrees with all given responses. Following round N, Bob reveals a minimizer $x_\star$ and a minimum value $f_\star$, again consistent with some $\mathtt{p}\in\mathtt{P}$. Alice then pays Bob $M(\mathtt{a},\mathtt{p})$.

In this game, if strong duality is attained between (6) and (8), i.e.,

$$\min_{\mathtt{a}\in\mathtt{A}} \max_{\mathtt{p}\in\mathtt{P}} M(\mathtt{a},\mathtt{p}) = \max_{\mathtt{p}\in\mathtt{P}} \min_{\mathtt{a}\in\mathtt{A}} M(\mathtt{a},\mathtt{p}),$$

then any attaining pair $(\mathtt{a},\mathtt{p})$ provides a saddle point. That is, $(\mathtt{a},\mathtt{p})$ forms a Nash equilibrium: if Alice plays $\mathtt{a}$ and Bob plays $\mathtt{p}$, neither player can strictly improve their payoff by deviating from their choice of $\mathtt{a}$ or $\mathtt{p}$, respectively.

In iterative games, subgame perfect equilibria refine Nash equilibria. Subgame perfect equilibrium asks that at every subgame state, neither player would deviate from their declared strategy. Definition 1 translates this idea to algorithm design. As a minor caveat, optimal subgame play is only required at histories $\mathcal{H}_n$ that the algorithm can produce. Unreachable histories are of lesser interest to us as, by definition, they will never arise when the method is applied.

4 Smooth Convex Problems

Smooth convex optimization is a well-studied area, making it a strong candidate for the development of subgame perfect methods. Indeed, subgame perfection was first attained for $\mathtt{A}_{\mathtt{span}}(N)$ and $\mathtt{P}_L$ by the authors in [10]. Here we present a technical sketch of this method's development as a process of "re-optimizing" the optimized gradient method.

The Optimized Gradient Method (OGM), first numerically observed by [8] and formally derived and analyzed by [16], maintains two iterate sequences x_n, z_{n+1} and a scalar sequence τ_n, as

$$(\tau_n, x_n, z_{n+1}) = \mathtt{OGM\_Update}(\tau_{n-1}, x_{n-1}, g_{n-1}, z_n)$$

with the update defined in Algorithm 1. Note the mild final step adjustment at iteration $n=N$ is necessary to be *exactly* minimax optimal.

Algorithm 1 OGM_Update($\hat{\tau}$, x, g, z)

1. $\tau_n = \begin{cases} \hat{\tau}+1+\sqrt{1+2\hat{\tau}} & \text{if } n<N \\ \hat{\tau}+\frac{1+\sqrt{1+4\hat{\tau}}}{2} & \text{if } n=N \end{cases}$
2. $x_n = \frac{\hat{\tau}}{\tau_n}(x-\frac{1}{L}g)+\frac{\tau_n-\hat{\tau}}{\tau_n}z$
3. $z_{n+1} = z-\frac{\tau_n-\hat{\tau}}{L}g$

One can understand the convergence of this method as maintaining the nonnegativity of a certain inductive hypothesis. Namely,

$$\begin{aligned} 0\le P_n :=& \tau_n\left(f_\star - f_n + \frac{1}{2L}\|g_n\|_2^2\right) \\ &+ \frac{L}{2}\|x_0-x_\star\|_2^2 - \frac{L}{2}\|z_{n+1}-x_\star\|_2^2. \end{aligned}$$

The final step adjustment makes the last inductive hypothesis

$$\begin{aligned} 0\le P_N :=& \tau_N\left(f_\star - f_N\right) \\ &+ \frac{L}{2}\|x_0-x_\star\|_2^2 - \frac{L}{2}\|z_{N+1}-x_\star\|_2^2, \end{aligned}$$

a simple rearrangement of which proves

$$\frac{f_N-f_\star}{\frac{L}{2}\|x_0-x_\star\|_2^2} \le \frac{1}{\tau_N}.$$

Hence, if one verifies $P_n\ge 0$ is inductively maintained, a $1/\tau_N$ convergence rate is proven. It turns out that proving this is surprisingly simple, as one has the identity

$$P_n = P_{n-1} + \tau_{n-1}Q_{n-1,n} + (\tau_n-\tau_{n-1})Q_{\star,n}$$

expressing P_n as a sum of nonnegative terms, proving its nonnegativity inductively. Here

$$Q_{i,j} := f_i - f_j - \langle g_j, x_i-x_j\rangle - \frac{1}{2L}\|g_i-g_j\|_2^2$$

is the cocoercivity inequality, which must be nonnegative for any L-smooth convex f. Note that $Q_{\star,n}$ uses $i=\star$ where $f_\star = f(x_\star)$ and $g_\star = 0$.

4.1 Considering the Adversary

Recall OGM inductively ensures $P_m \geq 0$, i.e.,

$$\frac{f_m - \frac{1}{2L}\|g_m\|_2^2 - f_\star}{\frac{L}{2}\|x_0 - x_\star\|_2^2} \leq \frac{1}{\tau_m}$$

for each past iteration $m = 0, \ldots, n-1$. For a given history $\mathcal{H}_n$, one may ask whether this worst-case upper bound can be attained. Otherwise, OGM's convergence theory (given the history) is needlessly conservative.

Our game-theoretic framing leads to a natural approach to answering this. The adversary may want to answer the following question: given a first-order history $\mathcal{H}_n$, *where can the adversary place the minimizer to make every inductive hypothesis bound from P_m as bad as possible?*

Formally, the adversary could solve

$$\max_{(f_\star, x_\star) \in R} \min_{0 \leq m \leq n-1} \frac{f_m - \frac{1}{2L}\|g_m\|_2^2 - f_\star}{\frac{L}{2}\|x_0 - x_\star\|_2^2}. \tag{11}$$

Here, R denotes the set of possible values of $f_\star$ and $x_\star$, and can be expressed explicitly as

$$R = \{(f_\star, x_\star) : P_i, Q_{\star,i} \geq 0 \text{ for } i = 0, \ldots, n-1\}.$$

Crucially, the adversary's optimization problem above is tractable to solve. Indeed, the set R is defined entirely by linear inequalities in $(f_\star, x_\star)$, and the overall problem can be expressed in terms of a second-order cone program.

We denote the optimal objective value of (11) by $1/\tau'$ and the optimal location of $x_\star$ by z'. If any iteration m has zero inductive hypothesis $P_m = 0$ at this $x_\star$, the resulting τ' and z' must match the values produced by OGM; that is, the optimal value is $1/\tau_m$, attained by z_{m+1}.

OGM updates x_n as a convex combination of a gradient descent step $x_{n-1} - \frac{1}{L}g_{n-1}$ and the hypothetical worst-case minimizer placement z_n (if $P_{n-1} = 0$). Hence, OGM hedges between gradient descent (greedy progress) and moving toward the anticipated worst-case point (conservatively mitigating its worst-case).

Given (11) is tractable to compute, one can immediately generalize this strategy. One could set $x_n = z'$ to mitigate the worst-case. However, doing so may be overly aggressive; the adversary is not obligated to place the minimizer at z'. Adopting the same hedged approach as OGM suggests an update of the form

$$x_n = \theta(x_m - \frac{1}{L}g_m) + (1-\theta)z'$$

for some m attaining the minimum in (11). This update matches OGM and other momentum methods in form while enabling dynamic runtime adaptation via the computation of z'.

Considering the dual problem to (11) allows us to analyze the resulting dynamic algorithm and optimally set the weight θ. Doing so provides us with an optimized inductive proof of convergence.

4.2 An Optimized Induction

Our goal is to improve on OGM by dynamically re-optimizing the inductive hypothesis at runtime as a function of the observed history $\mathcal{H}_n$. In particular, one can seek an inequality $P' \geq 0$ of the same inductive form as $P_{n-1} \geq 0$, replacing $(\tau_{n-1}, x_{n-1}, z_n)$ with values (τ', x_m, z').

As a family of inductive hypotheses, consider

$$P' = \sum_{i=0}^{n-1} \mu_i P_i + \sum_{i=0}^{n-1} \lambda_{\star,i} Q_{\star,i} + \epsilon \tag{12}$$

for any nonnegative $\mu_i, \lambda_{\star,i}, \epsilon$. Since the right-hand side is nonnegative, any P' of this form is a valid nonnegative hypothesis. We can find the optimal choice of the coefficients of this update by taking the dual to the problem (11).

As shorthand, let Z denote the matrix with columns $z_{i+1} - x_0$ and let G denote the matrix with columns g_i. Further, let $x_i^+ = x_i - \frac{1}{L}g_i$ denote gradient descent steps and $f_i^+ = f_i - \frac{1}{2L}\|g_i\|^2$ denote the objective upper bound from the descent lemma. Then, the dual to the problem in (11) is given by

$$\tau' = \begin{cases} \max & \sum_{i=0}^{n-1} \tau_i \mu_i + \sum_{i=0}^{n-1} \lambda_{\star,i} \\ \text{s.t.} & \epsilon(\mu, \lambda_\star) \geq 0 \\ & \mu, \lambda_\star \geq 0 \end{cases} \tag{13}$$

where the slack variable $\epsilon(\mu, \lambda_\star)$ is defined as

$$\begin{aligned} &\sum_{i=0}^{n-1} \mu_i \left[\tau_i (f_i^+ - f_m^+) + \frac{L}{2}\left(\|z_{i+1}\|_2^2 - \|x_0\|_2^2\right)\right] \\ &\quad + \sum_{i=0}^{n-1} \lambda_{\star,i} \left[f_i^+ - f_m^+ - \langle g_i, x_i^+ \rangle\right] \\ &\quad - \frac{L}{2}\left(\left\|x_0 + Z\mu - \frac{1}{L}G\lambda_\star\right\|_2^2 - \|x_0\|_2^2\right) \end{aligned}$$

with $m \in \arg\min_{i=0,\ldots,n-1} f_i^+$. Finally, the auxiliary iterate is set as

$$z' = x_0 + Z\mu - \frac{1}{L}G\lambda_\star. \tag{14}$$

4.3 The Subgame Perfect Gradient Method (SPGM)

By computing the above optimized inductive hypothesis $P' \geq 0$ at every iteration, one can then continue with OGM's update applied to (τ', x_m, g_m, z'), guaranteeing the core induction continues with

$$P_n = P' + \tau' Q_{m,n} + (\tau_n - \tau') Q_{\star,n}.$$

This process is formalized as the Subgame Perfect Gradient Method (SPGM):

Algorithm 2 SPGM_Update($\mathcal{H}_n$)

1. Compute the optimal $\mu, \lambda_\star$ via the SOCP (13).
2. Denote its optimal value τ' and z' as in (14).
3. $(\tau_n, x_n, z_{n+1}) =$ `OGM_Update`(τ', x_m, g_m, z').

Indeed, this method is subgame perfect.

Theorem 1 (Theorem 1, [10]). *SPGM is a subgame perfect method for reducing* $\frac{f(x_N)-f_\star}{\frac{L}{2}\|x_0-x_\star\|_2^2}$ *over all smooth convex problem instances* $\mathtt{P}_L$ *and* N*-step gradient span methods* $\mathtt{A}_{\mathtt{span}}(N)$.

The proof of this result proceeds by demonstrating matching minimax optimal algorithms and maximin hard problems for each subgame problem (9). The dynamic construction of this matching hard instance follows the style of hard instances designed by [5, 6]. The lower bounds are proven using the "zero-chain" style argument of Nemirovski and Yudin [21], where one argues that gradients can discover at most one coordinate per iteration while setting $d > N$.

4.4 Practical Variations

As written above, SPGM has several practical drawbacks. It requires the stored history/bundle of first-order information to grow linearly with iteration count and requires L as an input. Luckily, these can be resolved, leading to highly performant new algorithms.

Limiting Memory Extending SPGM and, in particular, the dynamically optimized inductive hypothesis (12) to operate with only a limited memory is straightforward. By retaining only the last k iterations, one can ask for the best hypothesis of the form

$$P' = \sum_{i=n-k}^{n-1} \mu_i P_i + \sum_{i=n-k}^{n-1} \lambda_{\star,i} Q_{\star,i} + \epsilon. \tag{15}$$

The resulting second-order cone problem for maximizing τ' in this case then has at most $2k$ variables. By restricting to the last k iterations, SPGM's per-iteration subproblem overhead becomes constant with respect to problem dimension and iteration count.

Adaptive Additions Widespread tools in first-order design (backtracking, restarting and preconditioning) were incorporated into SPGM by [19], yielding an adaptive method called ASPGM. On a wide test set of smooth convex problems, they found that the resulting parameter-free method is competitive with L-BFGS. While not uniformly surpassing quasi-Newton methods, ASPGM offers stronger nonasymptotic guarantees. Further, subgame perfect methods can provide more meaningful optimality certificates or stopping criteria due to their matching dual perspective. We refer numerically inclined readers to [19, Section 4].

4.5 Two Nonsmooth Extensions

The development of subgame perfect methods is not limited to smooth convex problems. In [12], we identified two subgame perfect methods for nonsmooth optimization.

If one assumes access to the proximal operator for a nonsmooth function, all of the above ideas can be generalized. The similarities of these two settings are well known since proximal steps can be interpreted as gradient steps on the Moreau envelope. A Subgame Perfect Proximal Point Algorithm (SPPPA) was proposed and analyzed in [12] following the same optimized induction approach as SPGM.

If one instead assumes access to only a subgradient oracle for problems in $\mathtt{P}_{M,D}$, an optimized dynamic method called the Kelley-Like cutting plane Method (KLM) was designed by Drori and Teboulle [7]. They proved KLM is minimax optimal and has dynamically improved convergence bounds. In [12], we designed exactly matching maximin hard instances, showing KLM is subgame perfect.

5 Open Frontiers

There is no shortage of domains to which these subgame perfect ideas can be extended. Within smooth convex problems, one may seek algorithms optimizing the final gradient norm or the average objective gap instead of the measures we focused on in this exposition. As for alternative problem settings, one can consider strongly convex or nonconvex problems, or settings in which only stochastic gradients or a monotone operator are provided as oracles. We expect a similarly perfect future for these areas.

Beyond such direct extensions, the subgame perfect framework enables new lines of inquiry. Below, we highlight three particular frontiers.

5.1 Designing Optimized Inductions

The perspective that led to the design of SPGM was dynamically optimizing the OGM inductive hypothesis at runtime. Such runtime adaptations (if done properly) can never worsen convergence guarantees. As a result, for any fixed-step method with a well-understood inductive proof, improved dynamic variants may readily follow. The recent line of work, starting with [23, 9], on gradient methods with memory pursued this with Nesterov-style estimate sequence proofs. There, Newton steps can dynamically improve the sequence's progress each iteration.

We are hopeful that the design philosophy of (re)optimizing inductions may be broadly useful.

5.2 Approximate Subgame Perfection

Subgame perfection provides a means to quantify the value of storing a bundle of older first-order information. Bundle methods have a long history of practical success leveraging histories, since [17, 20]. Subgame perfection may provide insights into the success of these methods. We may find that existing bundle methods are (approximately) subgame perfect, potentially under measures different from the final objective gap.

Conversely, new methods may arise by attempting to balance solutions to the subgame perfect design problem (9) and other practical considerations. In settings where efficient subgame perfect methods are not yet known (or may not exist), one may be able to solve approximations or relaxations of this design problem.

5.3 Optimality and Randomized Stepsizes for Gradient Descent

A recent line of work [3, 11, 29] considered optimizing the stepsizes used in gradient descent. These works prove accelerated $O(1/N^{1.2716\ldots})$ rates via fractal schedules. Given gradient descent iterates $x_{n+1} = x_n - h_n g_n/L$ for a vector of stepsizes $h \in \mathbb{R}^N$, the optimal stepsizes for smooth convex optimization are given by

$$\min_{h\in\mathbb{R}^N} \max_{\mathrm{p}\in\mathrm{P}_L} \frac{f(x_N) - f(x_\star)}{\frac{L}{2}\|x_0 - x_\star\|_2^2}. \tag{16}$$

This minimax problem was numerically solved for $N \leq 25$ by a branch-and-bound approach in [4]. Subsequently, the two works [11] and [29] provided analytic constructions of optimized stepsizes for all N that provably match (or beat) the numerical branch-and-bound solutions. For example, we conjecture that the OBS-F schedule of [11] attains (16).

Alas, we also conjecture that a minimax theorem does not hold for (16). That is, proving a matching $\Omega(1/N^{1.2716\ldots})$ lower bound may require developing hard instances as a function of h, rather than a single maximin hard instance.

One plausible path to avoid this obstruction lies in randomized stepsize policies. Success with randomization has already been seen in [2]. Often in game theory, such mixed strategies can restore the existence of equilibria. If a (randomized) minimax theory for gradient descent can be developed, this may open a path toward subgame perfect randomized gradient descent stepsizes.

References

[1] Peyman Afshani, Jérémy Barbay, and Timothy M. Chan. "Instance-Optimal Geometric Algorithms". In: *Journal of the ACM* 64.1 (2017). ISSN: 0004-5411. DOI: 10.1145/3046673. URL: https://doi.org/10.1145/3046673.

[2] Jason M. Altschuler and Pablo A. Parrilo. "Acceleration by Random Stepsizes: Hedging, Equalization, and the Arcsine Stepsize Schedule". In: *arXiv:2412.05790* (2024).

[3] Jason M. Altschuler and Pablo A. Parrilo. "Acceleration by stepsize hedging: Silver Stepsize Schedule for smooth convex optimization". In: *Mathematical Programming* (2024).

[4] Shuvomoy Das Gupta, Bart P. G. Van Parys, and Ernest K. Ryu. "Branch-and-bound performance estimation programming: a unified methodology for constructing optimal optimization methods". In: *Mathematical Programming* 204.1–2 (2024), 567–639. ISSN: 0025-5610. DOI: 10.1007/s10107-023-01973-1. URL: https://doi.org/10.1007/s10107-023-01973-1.

[5] Yoel Drori. "The exact information-based complexity of smooth convex minimization". In: *Journal of Complexity* 39 (2017), pp. 1–16.

[6] Yoel Drori and Adrien B Taylor. "On the oracle complexity of smooth strongly convex minimization". In: *Journal of Complexity* 68 (2022), p. 101590.

[7] Yoel Drori and Marc Teboulle. "An optimal variant of Kelley's cutting-plane method". In: *Mathematical Programming* 160 (2016), pp. 321–351.

[8] Yoel Drori and Marc Teboulle. "Performance of first-order methods for smooth convex minimization: A novel approach". In: *Mathematical Programming* 145.1-2 (2014), pp. 451–482.

[9] Mihai I. Florea and Yurii Nesterov. "An Optimal Lower Bound for Smooth Convex Functions". In: *Foundations of Computational Mathematics* 25.6 (2025), 1939–1973. ISSN: 1615-3375. DOI: 10.1007/s10208-025-09712-y. URL: https://doi.org/10.1007/s10208-025-09712-y.

[10] Benjamin Grimmer, Kevin Shu, and Alex L. Wang. "Beyond Minimax Optimality: A Subgame Perfect Gradient Method". In: *Mathematical Programming* (2026).

[11] Benjamin Grimmer, Kevin Shu, and Alex L. Wang. "Composing Optimized Stepsize Schedules for Gradient Descent". In: *Mathematics of Operations Research* (2025).

[12] Benjamin Grimmer and Alex L. Wang. "Subgame Perfect Methods in Nonsmooth Convex Optimization". In: *arXiv:2511.13639* (2025).

[13] Osman Güler. "New proximal point algorithms for convex minimization". In: *SIAM Journal on Optimization* 2.4 (1992), pp. 649–664.

[14] Uijeong Jang, Shuvomoy Das Gupta, and Ernest K. Ryu. "Computer-Assisted Design of Accelerated Composite Optimization Methods: OptISTA". In: *Mathematical Programming* (2025).

[15] Donghwan Kim. "Accelerated Proximal Point Method for Maximally Monotone Operators". In: *Mathematical Programming* 190.1–2 (2021), pp. 57–87.

[16] Donghwan Kim and Jeffrey A Fessler. "Optimized first-order methods for smooth convex minimization". In: *Mathematical Programming* 159.1 (2016), pp. 81–107.

[17] Claude Lemarechal. "An Extension of Davidon Methods to Nondifferentiable Problems". In: *Nondifferentiable Optimization*. Ed. by M. L. Balinski and Philip Wolfe. Berlin, Heidelberg: Springer Berlin Heidelberg, 1975, pp. 95–109. ISBN: 978-3-642-00764-4. DOI: 10.1007/BFb0120700. URL: https://doi.org/10.1007/BFb0120700.

[18] Felix Lieder. "On the Convergence Rate of the Halpern-iteration". In: *Optimization Letters* 15.2 (2021), pp. 405–418.

[19] Alan Luner and Benjamin Grimmer. "A Practical Adaptive Subgame Perfect Gradient Method". In: *arXiv:2510.21617* (2025).

[20] Robert Mifflin. "An algorithm for constrained optimization with semismooth functions". In: *Mathematics of Operations Research* 2.2 (1977), pp. 191–207.

[21] Arkadi Semenovič Nemirovski and David Borisovich Yudin. *Problem Complexity and Method Efficiency in Optimization*. Wiley-Interscience, 1983.

[22] Yurii Nesterov. "A method of solving a convex programming problem with convergence rate $O(1/k^2)$". In: *Soviet Mathematics Doklady* 27.2 (1983), pp. 372–376.

[23] Yurii Nesterov and Mihai I. Florea. "Gradient methods with memory". In: *Optimization Methods and Software* 37.3 (2022), pp. 936–953. DOI: 10.1080/10556788.2020.1858831. eprint: https://doi.org/10.1080/10556788.2020.1858831. URL: https://doi.org/10.1080/10556788.2020.1858831.

[24] Francesco Orabona. "A Modern Introduction to Online Learning". In: *arXiv:1912.13213* (2019).

[25] Jisun Park and Ernest K. Ryu. "Exact Optimal Accelerated Complexity for Fixed-Point Iterations". In: *International Conference on Machine Learning* (2022).

[26] Adrien B Taylor, Julien M Hendrickx, and François Glineur. "Exact worst-case performance of first-order methods for composite convex optimization". In: *SIAM Journal on Optimization* 27.3 (2017), pp. 1283–1313.

[27] Adrien B Taylor, Julien M Hendrickx, and François Glineur. "Smooth strongly convex interpolation and exact worst-case performance of first-order methods". In: *Mathematical Programming* 161.1–2 (2017), pp. 307–345.

[28] TaeHo Yoon, Ernest K. Ryu, and Benjamin Grimmer. "H-invariance theory: A complete characterization of minimax optimal fixed-point algorithms". In: *arXiv:2511.14915* (2025).

[29] Zehao Zhang and Rujun Jiang. "Accelerated Gradient Descent by Concatenation of Stepsize Schedules". In: *arXiv:2410.12395* (2024).

Chair's Column

In early 2026, the term of an exceptional team of SIAG/OPT officers came to an end. On behalf of our community, I would like to express my sincere gratitude for their dedication and service: Luís Nunes Vicente (Chair), Coralia Cartis (Vice Chair), Gabriele Eichfelder (Program Director), and Juliane Mueller (Secretary). Their leadership and commitment have had a lasting impact on SIAG/OPT, and we are deeply grateful for their contributions. I am delighted to be part of the new leadership team and greatly enjoy working alongside an outstanding group of officers: Frank E. Curtis (Vice Chair), Giacomo Nannicini (Program Director), and Ana Luísa Custódio (Secretary). The business meeting generated a number of valuable ideas and suggestions, several of which we have already begun to pursue.

As I write this, I have just returned from the 2026 SIAM Conference on Optimization in Edinburgh. The scientific energy of the meeting, the stimulating discussions, and the many opportunities to reconnect with colleagues and welcome new participants continue to resonate. Gathering with the international optimization community is always both a pleasure and a privilege. Having been part of it for more than thirty years, I find that the sense of coming home only grows stronger with the years.

A conference on the scale of SIAM OP26 could only succeed through the dedication of many people. I would like to thank the conference chairs—Miguel Anjos, Gabriele Eichfelder, and Luís Nunes Vicente—for their leadership and commitment, along with the organizing committee, the local organizing committee, all minisymposium organizers, and the SIAM staff, whose efforts made this wonderful meeting possible.

I would especially like to recognize the many helping hands from the University of Edinburgh. Their support behind the scenes contributed enormously to the smooth operation of the conference. On a personal note, Lars Schewe was my point of contact for several Auditorium events and, in tandem with the tech team, did a marvelous job. Thank you, Lars, and thank you to the entire Edinburgh team.

I am equally grateful to the speakers and participants. Their presentations, questions, and discussions created a lively and engaging atmosphere throughout the conference. The combination of high-quality talks, stimulating exchanges of ideas, and opportunities for informal interaction made SIAM OP26 a memorable and rewarding experience for all involved.

SIAM provided the following statistics for SIAM OP26, highlighting the remarkable scale of the meeting:

- Total attendance: 1,347
- Total number of minisymposia: 403 (including 19 that were cancelled)
- Total number of contributed sessions: 36

- Total number of plenary and prize talks: 9
- Total number of minitutorials: 2 (each consisting of 2 parts)
- Parallelism: up to 29 parallel sessions

Returning to the business meeting, one idea that received considerable interest was the development of online seminars or tutorials under the auspices of SIAG/OPT. Possibilities discussed included highlighting particularly suitable presentations from existing online seminar series with a SIAG/OPT label and inviting SIAM Fellows from our Activity Group to deliver online tutorials.

Participants also expressed concerns about steadily rising conference fees. We recognize the importance of this issue for many members and will continue to follow it closely.

Before closing, I would like to thank the editors of SIAG on Optimization Views and News. My sincere thanks go to Matt Menickelly and Jan Kronqvist for their ongoing work, and to Dmitriy Drusvyatskiy for his outstanding service as co-editor over the past years. As an Editor-in-Chief myself, I have a deep appreciation for the considerable effort required to produce a high-quality publication. Much of the work happens behind the scenes, but it is vital for keeping our community informed and connected. With Dima's term having come to an end, we are currently in the process of appointing a new co-editor and look forward to welcoming a new member to the editorial team. Thank you, Matt, Jan, and Dima, for your dedication to SIAG/OPT Views and News and to our community.

I wish you all a productive and inspiring summer, and thank you for your continued contributions to our vibrant SIAG/OPT community.

-Michael