analyzing optimization algorithms with integral quadratic constraints

Laurent Lessard, Andrew Packard, and Benjamin Recht
University of California, Berkeley
• Are joined by their arxiv category

• Controls made the SVD to SDP jump in the early 90s

• ML + Optimization perhaps now the synergistic duo

• There are many untapped analysis tools from controls
optimization (for big data?)

minimize \( f(x) \)
subject to \( x \in \Omega \)

- closely related cousin where \( P \) is a simple convex function: minimize \( f(x) + P(x) \)
- need algorithms that scale linearly (or sub-linearly) with dimension and data
- currently favored family are the first-order methods
gradient descent

$$x[k + 1] = x[k] - \alpha \nabla f(x[k])$$

for constrained optimization, use projected gradient descent

$$x[k + 1] = \Pi_\Omega (x[k] - \alpha \nabla f(x[k]))$$
acceleration/multistep

Gradient method akin to an ODE

\[ x[k + 1] = x[k] - \alpha \nabla f(x[k]) \]
\[ \dot{x} = -\nabla f(x) \]

To prevent oscillation, add a second order term

\[ \ddot{x} = -b\dot{x} - \nabla f(x) \]
\[ x[k + 1] = x[k] - \alpha \nabla f(x[k]) + \beta (x[k] - x[k - 1]) \]

Heavy ball method (constant \( \alpha, \beta \))

\[ x[k + 1] = y[k] - \alpha \nabla f(x[k]) \]
\[ y[k] = (1 + \beta)x[k] - \beta x[k - 1] \]

When \( f \) is quadratic, this is Chebyshev’s iterative method
canonical first order methods

- Gradient
  \[ x[k + 1] = x[k] - \alpha \nabla f(x[k]) \]

- Heavy Ball
  \[ x[k + 1] = y[k] - \alpha \nabla f(x[k]) \]
  \[ y[k] = (1 + \beta)x[k] - \beta x[k - 1] \]

- Nesterov
  \[ x[k + 1] = y[k] - \alpha \nabla f(y[k]) \]
  \[ y[k] = (1 + \beta)x[k] - \beta x[k - 1] \]

- each analyzed using specialized techniques
- what’s the right algorithm for my problem?
- are there other algorithms in this space that could be more effective for specific instances?
Control theory is the study of dynamical systems with inputs

\[ \xi[k + 1] = A\xi[k] + Bu[k] \]
\[ y[k] = C\xi[k] + Du[k] \]

Simplest case of such systems are linear systems
The Lur’e problem

- A linear dynamical system is connected in feedback with a nonlinearity.
- When do all trajectories converge to a fixed point?

\[
\begin{align*}
\xi[k + 1] &= A\xi[k] + Bu[k] \\
y[k] &= C\xi[k] + Du[k] \\
u[k] &= \Delta(y[k])
\end{align*}
\]
\[ \xi[k + 1] = A\xi[k] + Bu[k] \]
\[ y[k] = C\xi[k] + Du[k] \]
\[ u[k] = \nabla f(y[k]) \]

**Method**

- **Gradient**

\[
\begin{bmatrix}
A & B \\
C & D
\end{bmatrix} = \begin{bmatrix}
1 & -\alpha \\
1 & 0
\end{bmatrix}
\]

- **Heavy Ball**

\[
\begin{bmatrix}
A & B \\
C & D
\end{bmatrix} = \begin{bmatrix}
1 + \beta & -\beta \\
1 & 0
\end{bmatrix} \begin{bmatrix}
-\alpha \\
0
\end{bmatrix}
\]

- **Nesterov**

\[
\begin{bmatrix}
A & B \\
C & D
\end{bmatrix} = \begin{bmatrix}
1 + \beta & -\beta \\
1 & 0
\end{bmatrix} \begin{bmatrix}
-\alpha \\
0
\end{bmatrix}
\]
Gradient method

\[
\begin{bmatrix}
A & B \\
C & D
\end{bmatrix} = \begin{bmatrix}
1 & -\alpha \\
1 & 0
\end{bmatrix}
\]

\[
\xi[k + 1] = A\xi[k] + Bu[k] \\
y[k] = C\xi[k] + Du[k] \\
u[k] = \nabla f(y[k])
\]

\[
\xi[k + 1] = \xi[k] - \alpha u[k] \\
y[k] = \xi[k] \\
u[k] = \nabla f(y[k])
\]

\[
x[k + 1] = x[k] - \alpha \nabla f(x[k])
\]
\[
\begin{bmatrix}
A & B \\
C & D
\end{bmatrix}
= 
\begin{bmatrix}
1 + \beta & -\beta \\
1 & -\beta \\
1 + \beta & -\beta
\end{bmatrix}
\begin{bmatrix}
-\alpha \\
0 \\
0
\end{bmatrix}
\]

\[
\xi_1[k + 1] = (1 + \beta)\xi_1[k] - \beta\xi_2[k] - \alpha u[k]
\]

\[
\xi_2[k + 1] = \xi_1[k]
\]

\[
y[k] = (1 + \beta)\xi_1[k] - \beta\xi_2[k]
\]

\[
u[k] = \nabla f(y[k])
\]

\[
\xi_1[k + 1] = (1 + \beta)\xi_1[k] - \beta\xi_1[k - 1] - \alpha u[k]
\]

\[
y[k] = (1 + \beta)\xi_1[k] - \beta\xi_1[k - 1]
\]

\[
u[k] = \nabla f(y[k])
\]

\[
x[k + 1] = y[k] - \alpha \nabla f(y[k])
\]

\[
y[k] = (1 + \beta)x[k] - \beta x[k - 1]
\]
How do you prove an algorithm converges?  

**Step 1:** find a fixed point.

\[
\nabla f(x_*) = 0 \implies \begin{cases} 
y_* = x_* \\
u_* = 0 \\
\xi_* = A\xi_* \\
x_* = C\xi_* \end{cases}
\]
How do you prove an algorithm converges?

Step 2: prove all trajectories converge to the fixed point

Simple case: \( f(x) = \frac{1}{2}x^TQx - \rho^Tx \)

\[ \nabla f(x) = Qx - \rho \quad x_* = Q^{-1}\rho \]

\[ \xi[k + 1] - \xi_* = (A + BQC)(\xi[k] - \xi_*) \]

Necessary and sufficient condition is \( \rho(A + BQC) < 1 \)

\[ \lim_{k \to \infty} \|\xi[k] - \xi_*\|^{1/k} \leq \rho(A + BQC) \]
Gradient method

\[ \alpha = \frac{2}{L+m} \]

\[ \rho(A + BQC) \leq \frac{\kappa-1}{\kappa+1} \]

\[ \beta = \frac{4}{(\sqrt{L}+\sqrt{m})^2} \]

\[ \rho(A + BQC) \leq \left( \frac{\sqrt{\kappa-1}}{\sqrt{\kappa+1}} \right)^{1/2} \]

Heavy Ball

\[ \alpha = \frac{1}{L} \]

\[ \rho(A + BQC) \leq 1 - \frac{1}{\sqrt{\kappa}} \]

\[ \beta = \frac{\sqrt{\kappa-1}}{\sqrt{\kappa+1}} \]

Nesterov

\[ mI \preceq Q \preceq LI \]

\[ \kappa = \frac{L}{m} \]
Theorem: \( \rho(A) < \rho \) if and only if there exists 
\[ P \succeq 0 \text{ satisfying } A^T P A - \rho^2 P < 0 \]

Proof: If \( \rho(A) < \rho \), then 
\[ P = \sum_{k=0}^{\infty} \rho^{-2k} (A^T)^k A^k \]
exists and satisfies the desired LMI.

Conversely, assume the LMI has a solution and let \( \lambda \) be an eigenvalue with corresponding eigenvector \( \xi \). Then 
\[ \xi^T A^T PA\xi - \rho^2 \xi^T P \xi = (|\lambda|^2 - \rho^2)\xi^T P \xi < 0 \]
which implies \( |\lambda|^2 < \rho^2 \)
Theorem: \( \rho(A) < \rho \) if and only if there exists \( P \succeq 0 \) satisfying \( A^T P A - \rho^2 P < 0 \)

For dynamical systems, if \( \xi[k+1] = A\xi[k] \) the LMI implies
\[
\xi[k+1]^T P \xi[k+1] < \rho^2 \xi[k]^T P \xi[k]
\]

Iterating the recursion to \( k=0 \) gives
\[
\xi[k]^T P \xi[k] < \rho^{2k} \xi[0]^T P \xi[0]
\]

which in turn implies
\[
\|\xi[k]\| \leq \sqrt{\text{cond}(P)} \rho^k \|\xi_0\|
\]
Lyapunov functions

- $V(x) \geq 0$
- $V(x_*) = 0$
- $V(x[k]) < V(x[k - 1])$

- LMI characterization of stability parametrizes quadratic Lyapunov functions for the system
- This notion generalizes to nonlinear systems
Suppose there exists a $P > 0$ and matrix $M$ such that

$$
\begin{bmatrix}
y_1 - y_2 \\
\Delta(y_1) - \Delta(y_2)
\end{bmatrix}^T M \begin{bmatrix}
y_1 - y_2 \\
\Delta(y_1) - \Delta(y_2)
\end{bmatrix} \geq 0 \quad \text{for all } y_1, y_2
$$

Then \((\xi[k] - \xi^*)^T P (\xi[k] - \xi^*) \leq \rho^{2k} (\xi[0] - \xi^*)^T P (\xi[0] - \xi^*)\)
\[
\xi[k + 1] = A\xi[k] + Bu[k] \\
y[k] = C\xi[k] + Du[k] \\
u[k] = \Delta(y[k])
\]
and there exists a \( P \) such that
\[
\begin{bmatrix}
y_1 - y_2 \\
\Delta(y_1) - \Delta(y_2)
\end{bmatrix}^T 
M 
\begin{bmatrix}
y_1 - y_2 \\
\Delta(y_1) - \Delta(y_2)
\end{bmatrix} \geq 0 
\text{ for all } y_1, y_2
\]

\[
[A \ B]^T P \begin{bmatrix} A & B \end{bmatrix} - \begin{bmatrix} \rho^2 P & 0 \\ 0 & 0 \end{bmatrix} + \begin{bmatrix} C & D \\ 0 & I \end{bmatrix}^T M \begin{bmatrix} C & D \\ 0 & I \end{bmatrix} \preceq 0
\]

Multiply both sides by \( \begin{bmatrix} \xi[k] - \xi_* \\ u[k] - u_* \end{bmatrix} \)

\[
(\xi[k + 1] - \xi_*)^T P(\xi[k + 1] - \xi_*) - \rho^2 (\xi[k] - \xi_*)^T P(\xi[k] - \xi_*) \\
+ \begin{bmatrix} y[k] - y_* \\ u[k] - u_* \end{bmatrix}^T M \begin{bmatrix} y[k] - y_* \\ u[k] - u_* \end{bmatrix} \leq 0
\]
Gradient method

Sector QC
\[
\begin{bmatrix}
z_1 - z_2 \\
\nabla f(z_1) - \nabla f(z_2)
\end{bmatrix}^T
\begin{bmatrix}
-2mL l_d & (L + m) l_d \\
(L + m) l_d & 2l_d
\end{bmatrix}
\begin{bmatrix}
z_1 - z_2 \\
\nabla f(z_1) - \nabla f(z_2)
\end{bmatrix} \geq 0
\]

aka cocoercivity:
\[
\langle \nabla f(z_1) - \nabla f(z_2), z_1 - z_2 \rangle \geq \frac{1}{L} \| \nabla f(z_1) - \nabla f(z_2) \|^2
\]

Proposition: If \( f \) is convex, then \( f \) satisfies the Sector QC iff \( f \) has \( L \)-Lipschitz gradients and is strongly convex with parameter \( m \).
Gradient method

Sector QC

\[
\begin{bmatrix}
  z_1 - z_2 \\
  -\nabla f(z_1) - \nabla f(z_2)
\end{bmatrix}
\begin{bmatrix} -2mLl_d & (L + m)l_d \\ (L + m)l_d & 2l_d \end{bmatrix}
\begin{bmatrix}
  z_1 - z_2 \\
  \nabla f(z_1) - \nabla f(z_2)
\end{bmatrix} \geq 0
\]

\[
[A \ B]^T P [A \ B] - \begin{bmatrix}
  \rho^2 P & 0 \\
  0 & 0
\end{bmatrix} + \begin{bmatrix}
  C & D \\
  0 & I
\end{bmatrix}^T M \begin{bmatrix}
  C & D \\
  0 & I
\end{bmatrix} \preceq 0
\]

\[
\rho \begin{bmatrix}
  1 - \rho^2 & -\alpha \\
  -\alpha & \alpha^2
\end{bmatrix}
+ \mu \begin{bmatrix}
  -2mL & L + m \\
  L + m & -2
\end{bmatrix} \preceq 0
\]

Setting \( p=1 \), and setting the LMI to be exactly equal to zero, gives

\[
\rho = \frac{\kappa - 1}{\kappa + 1}
\]
The sector quadratic constraint is not sufficient to prove stability
Main Result (1): Suppose that there exists a linear system $\Psi$ and a matrix $M$ such that for any sequence $y_1, \ldots, y_T$

$$\sum_{k=1}^{T} \rho^{-2k} (z[k] - z^*)^T M (z[k] - z^*) \geq 0$$

and there exists a $P > 0$ such that

$$\begin{bmatrix} \hat{A} & \hat{B} \end{bmatrix}^T P \begin{bmatrix} \hat{A} & \hat{B} \end{bmatrix} - \begin{bmatrix} \rho^2 P & 0 \\ 0 & 0 \end{bmatrix} + \begin{bmatrix} \hat{C} & \hat{D} \end{bmatrix}^T M \begin{bmatrix} \hat{C} & \hat{D} \end{bmatrix} \leq 0$$

Then

$$\xi[k + 1] = A\xi[k] + Bu[k]$$
$$y[k] = C\xi[k] + Du[k]$$
$$u[k] = \nabla f(y[k])$$

$$\zeta[k + 1] = A\zeta[k] + B^{(u)}\zeta[k] + B^{(y)}y[k]$$
$$z[k] = C\zeta[k] + D^{(u)}u[k] + D^{(y)}y[k]$$

$\xi$ is a composite system matrices
**off-by-one IQC**

**Main Result (2):** Let $f$ be a strongly convex function with $L$-Lipschitz gradients and strong convexity parameter $m$. Then for any sequence $y[0], \ldots, y[T]$ with $u[k] = \nabla f(y[k])$

$$\sum_{k=1}^{T} \rho^{-2k} (u[k] - my[k])^T \left\{ L(y[k] - \rho^2 y[k-1]) - (u[k] - \rho^2 u[k-1]) \right\} \geq 0$$

- Without the delay terms ($\rho=0$), this is just the sector QC
- Builds on *Popov* and *Zames-Falb multipliers* from control.
- Elementary proof using co-coercivity inequalities.

$$\sum_{k=1}^{T} \rho^{-2k} (z[k] - z_*)^T M(z[k] - z_*) \geq 0$$

$$\begin{bmatrix} A_{\psi} & B_{\psi} \\ C_{\psi} & D_{\psi} \end{bmatrix} = \begin{bmatrix} 0 & \rho L l_d & \rho l_d \\ -\rho l_d & L l_d & -l \\ 0 & -m l_d & l_d \end{bmatrix} \quad M = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$$
Nesterov

Heavy Ball

Convergence rate $\rho$

Condition number $\kappa = L/m$

- Predicted Nesterov rate
- Optimal Nesterov rate
- Optimal Gradient rate

- Predicted Heavy-ball rate
- Quadratic Heavy-ball rate
- Gradient rate
Nesterov Heavy Ball

\[ \kappa = \frac{3}{T} \]

Convergence rate \( \rho \)

Condition number \( \kappa = \frac{L}{m} \)

Graph showing the convergence rate for different conditions.
Heavy Ball isn’t stable

- **Aizerman’s conjecture** [1949]. A linear system in feedback with a sector nonlinearity is stable if the linear system is stable for any linear gain of the sector.

- **THE AIZERMAN CONJECTURE IS FALSE** [Krasovskii 1952]

- This is a very simple counterexample.

\[
\begin{align*}
f(x) &= \begin{cases} 
16x^2 + 90x + 135 & x < -3 \\
x^2 & x \in [-3, 0] \\
16x^2 & x \geq 0 
\end{cases} \\
m = 1 & \quad L = 16
\end{align*}
\]

If you start at \(x_0 \in [1.9, 2.4]\), Heavy Ball with standard parameters converges to the limit cycle.
Nesterov

Rate

Iterations (-log^{-1} \rho)

Iterations differ from the quadratic case by less than a factor of 2.
Heavy-Ball

**Rate**

Fix $\alpha = 1/L$.

Grid search over $\beta$ to find minimal convergence rate $\rho$
Integral Quadratic Constraints in Context

- Proposed by Megretski and Rantzer in 1996 (frequency domain)
- Generalizes the KYP Lemma/dissipativity theory
- Special case of S-Procedure/sum-of-squares hierarchy
- Drori and Teboulle 2013 used all quadratic constraints between time points to provide sharp analysis of gradient method for weakly convex functions.
- IQCs allow analysis which is dimension-free and certificates of size independent of the time horizon.
Extensions

Proximal/Projected methods

Achieve same rate as unconstrained case via an LFT argument

Removing strong convexity

Achieve standard $\tilde{O}(\text{poly}(k^{-1}))$ rates by adding a regularization term
Noisy Gradients

\[ u[k] = \nabla f(y[k]) + \omega[k] \]
\[ ||\omega[k]|| \leq \delta ||\nabla f(y[k])|| \]

Gradient method becomes robust when \( \alpha = 1/L \)
Synthesis (brutal forces)

• test all algorithms with two states
• parameterization in terms of \((\alpha, \beta_1, \beta_2)\):
\[
x_{k+1} = x_k - \alpha \nabla f(x_k + \beta_2(x_k - x_{k-1})) + \beta_1(x_k - x_{k-1})
\]

Special cases:
- Gradient: \((\alpha, \beta_1, \beta_2) = (\alpha, 0, 0)\)
- Heavy Ball: \((\alpha, \beta_1, \beta_2) = (\alpha, \beta, 0)\)
- Nesterov: \((\alpha, \beta_1, \beta_2) = (\alpha, \beta, \beta)\)
Synthesis (brutal forces)

- parameterization in terms of \((\alpha, \beta_1, \beta_2)\):
  \[ x_{k+1} = x_k - \alpha \nabla f(x_k + \beta_2(x_k - x_{k-1})) + \beta_1(x_k - x_{k-1}) \]

- Faster than the gradient method AND provably robust to noise.
- Suggests that more sophisticated algorithm design is possible.
Conclusions

- IQCs provide a powerful proof system for algorithm analysis by replacing complicated nonlinearities with quadratic constraint sets.

- *Collects constraints about function classes, not algorithms.*

- New proofs of convergence for popular first-order methods.

- Enables numerical exploration of parameter spaces.

- Only beginning to get a sense of what IQCs can tell us about optimization schemes.

- Many more control theory techniques that may provide new insight when applied to optimization and machine learning.
Open Problems

- Improve the analysis for Nesterov’s method using refined IQCs
- An analytic proof of Nesterov’s method using IQCs
- Lower bounds using Zames-Falb IQCs and Megretski argument
- Integrating time varying plants. Is Nonlinear Conjugate Gradient actually stable?
- Is there a way to choose the stepsize using adaptive control techniques?
- New algorithm design via DK iterations and IQC-based nonlinear control synthesis.
- Stochastic coordinate descent and stochastic gradient descent via expected IQCs
- Subgaussian noise analysis via LQG and Ricatti equations
- Bringing the function value into the picture. The function itself is Lyapunov!
- Extending the library of IQCs.
- Automatically proving and deriving IQCs via sum-of-squares techniques
- Smaller function classes. With more structure, do we get better rates?
- Search for non-quadratic Lyapunov functions using IQC + SOS
- Analyzing really complicated interconnections for modular machine learning
Acknowledgments

Laurent Lessard          Andrew Packard

• Many thanks to Elad Hazan, Ali Jadbabaie, Pablo Parrilo and Peter Seiler for many helpful pointers and discussions

• Preprint at arxiv.org/abs/1408.3595

Thanks!