Solving Second-Order Cone Programs in Matrix Multiplication Time

Background

Second-Order Cone Programming (SOCP) provides a generalized representation of several well-known convex optimization problems.

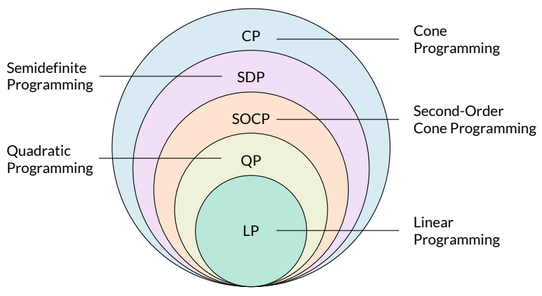


Figure 1. A hierarchy of the optimization problems, with LP being the easiest and CP being the most general form.

Definition: Second-Order Cone

A second-order cone $\mathcal{L}^k$ is defined as

$$\{(x_0, \widetilde{\boldsymbol{x}}), \widetilde{\boldsymbol{x}} \in \mathbb{R}^k : \|\widetilde{\boldsymbol{x}}\|_2 \leq x_0\}.$$

A second-order cone program is a convex optimization problem with:

- **Objective function:** Linear function $\boldsymbol{c}^\top \boldsymbol{x}$
- **Constraint function:** Intersection of an affine set $\mathbf{A}\boldsymbol{x} = \boldsymbol{b}$ and the Cartesian product $\mathcal{L}$ of second-order cones.

Definition: Second-Order Cone Program (SOCP)

Given the constraint matrix $\mathbf{A} \in \mathbb{R}^{m \times n}$, two vectors $\boldsymbol{b} \in \mathbb{R}^m$ and $\boldsymbol{c} \in \mathbb{R}^n$, and r second-order cones $\mathcal{L}_1, \ldots, \mathcal{L}_r$, the optimization problem can be expressed as:

$$\min \boldsymbol{c}^\top \boldsymbol{x} \text{ subject to } \mathbf{A}\boldsymbol{x} = \boldsymbol{b}, \boldsymbol{x}_i \in \mathcal{L}_i \text{ for all } i \in [r],$$

where $\boldsymbol{x}$ is the concatenation of $\boldsymbol{x}_i$ lying inside the domain $\mathcal{L} \stackrel{\text{def}}{=} \mathcal{L}_1 \times \cdots \times \mathcal{L}_r$ and each $\mathcal{L}_i \in \mathbb{R}^{n_i}$ is a second-order cone.

Significance of SOCP

Second-order cone programs provide a **general framework** for solving a wide range of linear and quadratic programming problems with **applications** in:

- Financial portfolio optimization
- Engineering and control systems
- Energy management systems
- Logistics and supply chain management
- Machine learning algorithms, such as support vector machines

Research Objective

Develop the current fastest SOCP algorithm and mathematically prove its convergence across all SOCP problems.

Previous Work

Table 1. Previous work using interior point methods in various convex problems.

Year	Type of Convex Program	Complexity Achieved	Researchers
1984	Linear Program	$O(n^{3.5})$	Karmarkar
1994	Second-Order Cone Program	$O(n^{\omega+0.5})$	Nesterov, Nemirovski
2019	Constant Dimension Convex Program	$O(n^{\omega})$	Lee, Song, Zhang
2021	Linear Program	$O(n^{\omega})$	Cohen, Lee, Song
2023	Quadratic Program	$O(n^{\omega})$	Gu, Song, Zhang
2023	**Second-Order Cone Program**	$O(n^{\omega})$	**This research**

Research Gap and My Approach

Challenges

- The highly complex structure of SOCP makes it difficult to find efficient algorithms.
- The sizes of the block matrices corresponding to constraints can vary significantly; high-dimensional blocks create bottlenecks in SOCP algorithms.
- Heuristic algorithms may not converge stably in all cases.

My Approach

- Identified **approximation techniques** to overcome the complexity barrier inherent in SOCP algorithms.
- Developed a novel approach to **decompose** large cone constraints into smaller ones, overcoming the bottleneck.
- **Mathematically proved** that the proposed algorithm **converges** in matrix multiplication time by finding bounds and utilizing self-concordance properties.

Breakthrough 1: Approximation Technique

To reduce the cost of computing $(\boldsymbol{x}, \boldsymbol{s})$ at each iteration, I developed a technique to approximate $(\boldsymbol{x}, \boldsymbol{s})$. Selective blocks of the approximation $(\bar{\boldsymbol{x}}, \bar{\boldsymbol{s}})$ are updated at each step to ensure it remains close to the central path.

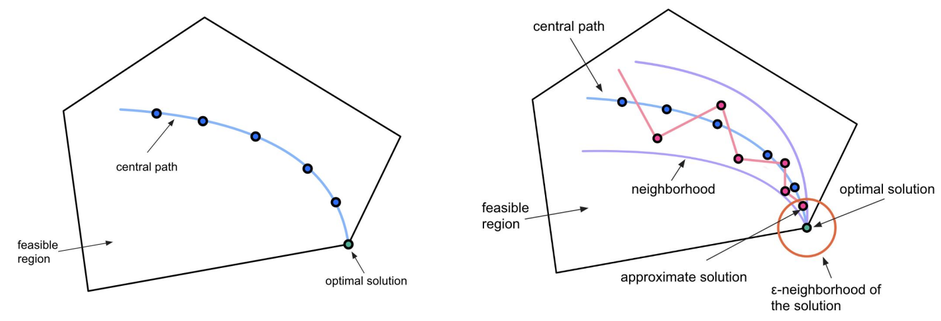
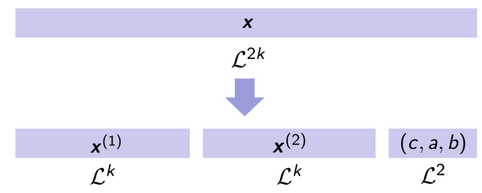


Figure 2. Approximation iterates near central path and converges to optimum.

Breakthrough 2: Cone Splitting

Updating a high-dimension block is still expensive when its dimension n_i is large. I developed a novel technique to transform high-dimension cones into the intersection of smaller cones and an affine space:



where $a = \boldsymbol{x}_0^{(1)}$ and $b = \boldsymbol{x}_0^{(2)}$

Figure 3. Illustration of the cone splitting technique. In this example, the cone $\mathcal{L}^{2k}$ is split into three smaller cones $\mathcal{L}^k$, $\mathcal{L}^k$, and $\mathcal{L}^2$ with the linear constraints $a = \boldsymbol{x}_0^{(1)}$ and $b = \boldsymbol{x}_0^{(2)}$ added.

My Main Theorem

Given a second-order cone program with full-rank constraint matrix $\mathbf{A} \in \mathbb{R}^{m \times n}$ and $m \leq n$, define the following parameters:

1. Outer radius R: Assume $\|\mathbf{A}\|_1 \leq R$, and $\|\boldsymbol{b}\|_2 \leq R$ and any feasible solution $\boldsymbol{x}$ satisfies $\|\boldsymbol{x}\|_2 \leq R$
2. Lipschitz constant L: $\|\boldsymbol{c}\|_2 \leq L$.

Then, for any $0 < \epsilon \leq 1/2$, there is a deterministic algorithm which runs in time

$$\widetilde{O}((n^\omega + n^{2.5-\alpha/2+o(1)} + n^{2+o(1)} r^{1/6}) \log(\tfrac{R}{\varepsilon}))$$

and outputs a solution $\boldsymbol{x} \in \mathcal{L}$ such that

$$\|\mathbf{A}\boldsymbol{x} - \boldsymbol{b}\|_1 \leq \epsilon$$

and

$$\boldsymbol{c}^\top \boldsymbol{x} \leq \min_{\mathbf{A}\boldsymbol{x}=\boldsymbol{b}, \boldsymbol{x}_i \in \mathcal{L}_i \text{ for all } i \in [r]} \boldsymbol{c}^\top \boldsymbol{x} + \epsilon L.$$

Applications and Impact

The new SOCP solution can be applied to significantly improve efficiency in:

Financial Portfolio Optimization

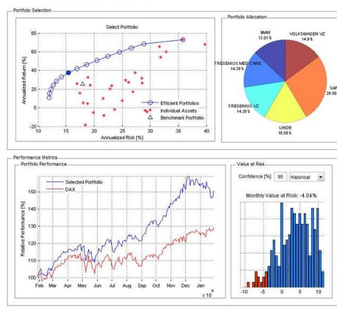


Image by Diane-Laure Arjaliès

Support Vector Machines

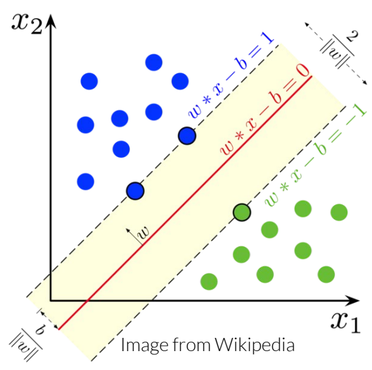


Image from Wikipedia

AC Optimal Power Flow

Image from GE.com

Key Findings and Conclusions

- **New Strategies to Reduce Runtime:** Developed a novel approach that decomposes large constraints into smaller ones to address the high computational costs of SOCP.
- **Academic Significance:** Mathematically proved that the new algorithm converges in $O(n^\omega)$ time, **reaching the theoretical limit**.
- **SOCP Solver Implementation:** Developed a software SOCP solver ready for use by the research community.
- **Applications Across Industries:** Explored various applications of my solution and showed that it achieves significant efficiency improvement across a variety of domains.

Future Work

- Specialize the new SOCP solution to achieve further performance improvements for specific applications.
- Apply a similar cone splitting idea to other types of conic programming, such as semidefinite programming.

All images created by the researcher unless otherwise noted.