A Computationally Viable Numerical Gradient-based Technique for Optimal Covering Problems

Gokul Rajaraman[1] Debasish Chatterjee[2]

[1]Department of Mechanical Engineering
Indian Institute of Technology, Bombay

[2]Center for Systems and Control Engineering
Indian Institute of Technology, Bombay

NeurIPS 2025

The Optimal Covering Problem

Optimal covering using Euclidean balls

Given a compact set $\mathcal{M} \subset \mathbb{R}^N$, find n balls of minimum radius that cover $\mathcal{M}$.

$$\epsilon(\mathcal{M}, n) := \min_{(x_1,\ldots,x_n)\in\mathbb{R}^{N\times n}} \max_{y\in\mathcal{M}} \min_{i=1,\ldots,n} \|y - x_i\| \tag{1}$$

Solving (1) yields an optimal n-point representation (a Chebyshev n-net) of $\mathcal{M}$.

Significance

Some applications across diverse fields:

- **Function Learning:** Approximation/Optimization of a Lipschitz function with error bounds

Significance

Some applications across diverse fields:

- **Function Learning:** Approximation/Optimization of a Lipschitz function with error bounds
- **Information Theory:** Optimal signal representation given error tolerance

Significance

Some applications across diverse fields:

- **Function Learning:** Approximation/Optimization of a Lipschitz function with error bounds
- **Information Theory:** Optimal signal representation given error tolerance
- **Operations Research:** Given a budget of n 'facilities' (e.g., sensors, gas stations, cell towers), position them optimally while ensuring coverage

Significance

Some applications across diverse fields:

- **Function Learning:** Approximation/Optimization of a Lipschitz function with error bounds
- **Information Theory:** Optimal signal representation given error tolerance
- **Operations Research:** Given a budget of n 'facilities' (e.g., sensors, gas stations, cell towers), position them optimally while ensuring coverage

Challenge: NP-hard even for $n = 1$ (Chebyshev center problem)

The Challenge

Why is this difficult?

- Nonconvex, nonsmooth objective $\min_{i=1,\ldots,n} \|y - x_i\|$

The Challenge

Why is this difficult?

- Nonconvex, nonsmooth objective $\min_{i=1,\ldots,n} \|y - x_i\|$
- No analytical expression for inner max-min.

The Challenge

Why is this difficult?

- Nonconvex, nonsmooth objective $\min_{i=1,\ldots,n} \|y - x_i\|$
- No analytical expression for inner max-min.
- Examples of existing 'zeroth-order' methods:

The Challenge

Why is this difficult?

- Nonconvex, nonsmooth objective $\min_{i=1,\ldots,n} \|y - x_i\|$
- No analytical expression for inner max-min.
- Examples of existing 'zeroth-order' methods:
  - Simulated Annealing via Markov Chain Monte Carlo

The Challenge

Why is this difficult?

- Nonconvex, nonsmooth objective $\min_{i=1,\ldots,n} \|y - x_i\|$
- No analytical expression for inner max-min.
- Examples of existing 'zeroth-order' methods:
  - Simulated Annealing via Markov Chain Monte Carlo
  - Simplicial algorithms (e.g., Nelder-Mead)

The Challenge

Why is this difficult?

- Nonconvex, nonsmooth objective $\min_{i=1,\ldots,n} \|y - x_i\|$
- No analytical expression for inner max-min.
- Examples of existing 'zeroth-order' methods:
  - Simulated Annealing via Markov Chain Monte Carlo
  - Simplicial algorithms (e.g., Nelder-Mead)
- Regularity of the objective not exploited $\rightarrow$ slow convergence and poor scalability

The Challenge

Why is this difficult?

- Nonconvex, nonsmooth objective $\min_{i=1,\ldots,n} \|y - x_i\|$
- No analytical expression for inner max-min.
- Examples of existing 'zeroth-order' methods:
  - Simulated Annealing via Markov Chain Monte Carlo
  - Simplicial algorithms (e.g., Nelder-Mead)
- Regularity of the objective not exploited $\rightarrow$ slow convergence and poor scalability
- Numerical gradient-based methods: better scaling, more efficient

Reformulation

Step 1: Restrict $(x_1, \ldots, x_n)$**:** All global optimizers of (1) can be restricted to a ball of radius $2\epsilon(\mathcal{M}, 1)$, centered at the Chebyshev center (denoted by $\mathbb{B}(\mathcal{M})$)

Reformulation

Step 1: Restrict $(x_1, \ldots, x_n)$: All global optimizers of (1) can be restricted to a ball of radius $2\epsilon(\mathcal{M}, 1)$, centered at the Chebyshev center (denoted by $\mathbb{B}(\mathcal{M})$)

Step 2: Reformulate using slack variable t:

$$\epsilon(\mathcal{M}, n)^2 = \min_{(x_1,\ldots,x_n)\in\mathbb{B}(\mathcal{M})^n} \mathcal{G}(x_1, \ldots, x_n)$$

where $\mathcal{G}(x_1, \ldots, x_n) := \max\{t \mid (t, y) \in \mathbb{R} \times \mathcal{M}, t - \|y - x_i\|^2 \leq 0 \text{ for all } i\}$

Reformulation

Step 1: Restrict $(x_1, \ldots, x_n)$: All global optimizers of (1) can be restricted to a ball of radius $2\epsilon(\mathcal{M}, 1)$, centered at the Chebyshev center (denoted by $\mathbb{B}(\mathcal{M})$)

Step 2: Reformulate using slack variable t:

$$\epsilon(\mathcal{M}, n)^2 = \min_{(x_1,\ldots,x_n)\in\mathbb{B}(\mathcal{M})^n} \mathcal{G}(x_1, \ldots, x_n)$$

where $\mathcal{G}(x_1, \ldots, x_n) := \max\{t \mid (t, y) \in \mathbb{R} \times \mathcal{M}, t - \|y - x_i\|^2 \leq 0 \text{ for all } i\}$

Step 3: Represent $\mathcal{M}$ via smooth constraints
Assume $\mathcal{M} = \{y \in \mathbb{R}^N \mid \varphi_j(y) \leq 0, j = 1, \ldots, k\}$ with φ_j continuously differentiable

Main Theoretical Contribution

$\mathcal{G}(x_1, \ldots, x_n)$ is the optimal value (marginal) function of a nonconcave maximization. Can we comment on regularity properties of $\mathcal{G}$?

Main Theoretical Contribution

$\mathcal{G}(x_1, \ldots, x_n)$ is the optimal value (marginal) function of a nonconcave maximization. Can we comment on regularity properties of $\mathcal{G}$?

Theorem (Lipschitz Continuity of $\mathcal{G}$)

Under Mangasarian-Fromovitz Constraint Qualification (MFCQ), the marginal function $\mathcal{G}$ is **locally Lipschitz**, *and its restriction to $\mathbb{B}(\mathcal{M})^n$ is L-Lipschitz.*

Main Theoretical Contribution

$\mathcal{G}(x_1, \ldots, x_n)$ is the optimal value (marginal) function of a nonconcave maximization. Can we comment on regularity properties of $\mathcal{G}$?

Theorem (Lipschitz Continuity of $\mathcal{G}$)

Under Mangasarian-Fromovitz Constraint Qualification (MFCQ), the marginal function $\mathcal{G}$ is **locally Lipschitz**, *and its restriction to $\mathbb{B}(\mathcal{M})^n$ is L-Lipschitz.*

- Derived using results from sensitivity analysis of NLPs (Fiacco & Ishizuka, 1990)

Main Theoretical Contribution

$\mathcal{G}(x_1, \ldots, x_n)$ is the optimal value (marginal) function of a nonconcave maximization. Can we comment on regularity properties of $\mathcal{G}$?

Theorem (Lipschitz Continuity of $\mathcal{G}$)

Under Mangasarian-Fromovitz Constraint Qualification (MFCQ), the marginal function $\mathcal{G}$ is **locally Lipschitz**, *and its restriction to $\mathbb{B}(\mathcal{M})^n$ is L-Lipschitz.*

- Derived using results from sensitivity analysis of NLPs (Fiacco & Ishizuka, 1990)
- Enables use of numerical gradient-based algorithms

Main Theoretical Contribution

$\mathcal{G}(x_1, \ldots, x_n)$ is the optimal value (marginal) function of a nonconcave maximization. Can we comment on regularity properties of $\mathcal{G}$?

Theorem (Lipschitz Continuity of $\mathcal{G}$)

Under Mangasarian-Fromovitz Constraint Qualification (MFCQ), the marginal function $\mathcal{G}$ is **locally Lipschitz**, *and its restriction to $\mathbb{B}(\mathcal{M})^n$ is L-Lipschitz.*

- Derived using results from sensitivity analysis of NLPs (Fiacco & Ishizuka, 1990)
- Enables use of numerical gradient-based algorithms
- First result exploiting the regularity of the inner problem for optimal covering

Our Algorithm: gradOptNet

gradOptNet **performs a zeroth-order stochastic projected gradient descent**

Our Algorithm: gradOptNet

`gradOptNet` **performs a zeroth-order stochastic projected gradient descent**
Input: Initial point $x_0 \in \mathbb{B}(\mathcal{M})^n$, stepsize $\gamma > 0$, smoothing parameter δ, iteration number $T \geq 1$, and parameter b. The following outlines `gradOptNet`'s update step:

Our Algorithm: gradOptNet

gradOptNet **performs a zeroth-order stochastic projected gradient descent**
Input: Initial point $x_0 \in \mathbb{B}(\mathcal{M})^n$, stepsize $\gamma > 0$, smoothing parameter δ, iteration number $T \geq 1$, and parameter b. The following outlines gradOptNet's update step:

- Sample $w_{1,t}, \ldots, w_{b_1,t}$ uniformly from the unit sphere in $\mathbb{R}^{N \times n}$.

Our Algorithm: gradOptNet

gradOptNet **performs a zeroth-order stochastic projected gradient descent**
Input: Initial point $x_0 \in \mathbb{B}(\mathcal{M})^n$, stepsize $\gamma > 0$, smoothing parameter δ, iteration number $T \geq 1$, and parameter b. The following outlines gradOptNet's update step:

- Sample $w_{1,t}, \ldots, w_{b_1,t}$ uniformly from the unit sphere in $\mathbb{R}^{N \times n}$.
- Let $g_{i,t} = \frac{Nn}{2\delta} \left(\mathcal{G}(x_t + \delta w_{i,t}) - \mathcal{G}(x_t - \delta w_{i,t}) \right) w_{i,t}$ for each $i \in \{1, \ldots, b\}$.

Our Algorithm: gradOptNet

`gradOptNet` **performs a zeroth-order stochastic projected gradient descent**
Input: Initial point $x_0 \in \mathbb{B}(\mathcal{M})^n$, stepsize $\gamma > 0$, smoothing parameter δ, iteration number $T \geq 1$, and parameter b. The following outlines `gradOptNet`'s update step:

- Sample $w_{1,t}, \ldots, w_{b_1,t}$ uniformly from the unit sphere in $\mathbb{R}^{N \times n}$.
- Let $g_{i,t} = \frac{Nn}{2\delta} \left(\mathcal{G}(x_t + \delta w_{i,t}) - \mathcal{G}(x_t - \delta w_{i,t}) \right) w_{i,t}$ for each $i \in \{1, \ldots, b\}$.
- Set $v_t = \frac{1}{b} \sum_{i=1}^{b} g_{i,t}$. Update $x_{t+1} = \mathrm{Proj}_{\mathbb{B}(\mathcal{M})^n} (x_t - \gamma v_t)$.

Our Algorithm: gradOptNet

`gradOptNet` **performs a zeroth-order stochastic projected gradient descent**
Input: Initial point $x_0 \in \mathbb{B}(\mathcal{M})^n$, stepsize $\gamma > 0$, smoothing parameter δ, iteration number $T \geq 1$, and parameter b. The following outlines `gradOptNet`'s update step:

- Sample $w_{1,t}, \ldots, w_{b_1,t}$ uniformly from the unit sphere in $\mathbb{R}^{N \times n}$.
- Let $g_{i,t} = \frac{Nn}{2\delta} \left(\mathcal{G}(x_t + \delta w_{i,t}) - \mathcal{G}(x_t - \delta w_{i,t}) \right) w_{i,t}$ for each $i \in \{1, \ldots, b\}$.
- Set $v_t = \frac{1}{b} \sum_{i=1}^{b} g_{i,t}$. Update $x_{t+1} = \mathrm{Proj}_{\mathbb{B}(\mathcal{M})^n} (x_t - \gamma v_t)$.

Complexity Bound

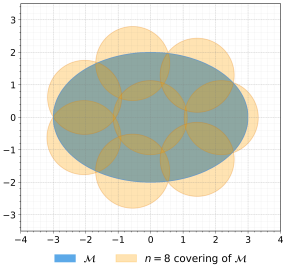
Requires $\mathcal{O}\left(N^{3/2} n^2 \epsilon(\mathcal{M}, 1) L^3 \delta^{-1} \varepsilon^{-3}\right)$ evaluations of $\mathcal{G}$ to obtain a $(\gamma, \delta, \varepsilon)$-generalized Goldstein stationary point.

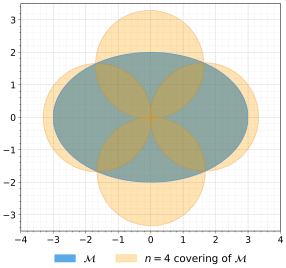
Numerical Experiments

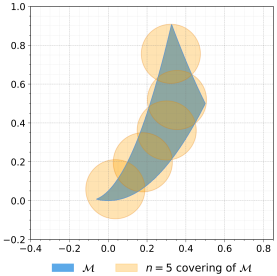
Setup:

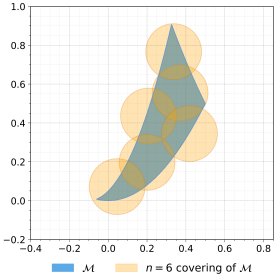
- For the ease of visualization, we present `gradOptNet` results in $\mathbb{R}^2$ with varying n
- Inner problem solved using `BARON` (good global convergence)
- Comparison (speed and accuracy) done with simulated annealing

Results: Visualization of Covering









NEURAL INFORMATION PROCESSING SYSTEMS

Results: gradOptNet vs Simulated Annealing & Ground Truth

n	gradOptNet	$\epsilon(\mathcal{M}, n)$	Gap (%)↓
2	1.00203	1	0.20
3	0.868089	0.866025	0.24
4	0.712492	0.707107	0.76
5	0.612386	0.609382	0.49
6	0.562335	0.555905	1.16
7	0.508007	0.5	1.60
8	0.448582	0.445041	0.80
9	0.423327	0.414213	2.20
10	0.413452	0.394930	4.69
11	0.385371	0.380083	1.39
12	0.372811	0.361141	3.23

gradOptNet vs ground truth for 2-D unit disk

Method	$\epsilon(\mathcal{M}, 4) \downarrow$			**Evals** $\downarrow$
	Trial 1	**Trial 2**	**Trial 3**	
gradOptNet	1.682	1.706	1.698	4800
Sim. Annealing	2.227	2.574	2.301	10000

gradOptNet vs Simulated Annealing (ellipse, $n = 4$)

Future Directions

- Accelerate evaluation of inner problem for convex sets (DC programming)
- Exploit permutation invariance of $\mathcal{G}(x_1, \ldots, x_n)$ to restrict search space
- Scaling to high dimensions and large n remains a problem due to nonconvexity

References

[1] A. R. Alimov and I. G. Tsar'kov.
Geometric Approximation Theory.
Springer Monographs in Mathematics. Springer, Cham, 2021.

[2] A. V. Fiacco and Y. Ishizuka.
Sensitivity and stability analysis for nonlinear programming.
Annals of Operations Research, 27(1):215–235, 1990.

[3] T. Lin, Z. Zheng, and M. I. Jordan.
Gradient-free methods for deterministic and stochastic nonsmooth nonconvex optimization.
Advances in Neural Information Processing Systems, volume 35, 2022.

[4] P. Paruchuri and D. Chatterjee.
Attaining the Chebyshev bound for optimal learning: A numerical algorithm.
Systems & Control Letters, 181:105648, 2023.