Multiple-response agents: Fast, feasible, approximate primal recovery for dual optimization methods

Tetiana Parshakova[♣], Yicheng Bai[♣]
Garrett van Ryzin[♣], Stephen Boyd[⋆]
[♣]Amazon SCOT, [⋆]Stanford

Distributed convex optimization

$$\begin{array}{ll} \text{minimize} & f(x) = \sum_{i=1}^K f(x_i) \\ \text{subject to} & \sum_{i=1}^K A_i x_i \leq b \end{array}$$

- $f: \mathbf{R}^{n_i} \to \mathbf{R} \cup \{\infty\}$ is closed, convex, and proper
- $A = (A_1, \ldots, A_K) \in \mathbf{R}^{m \times n}$ and $b \in \mathbf{R}^m$ given
- conjugate subgradient oracle

$$x_i(y_i) \in \underset{z_i \in \textbf{dom}\, f_i}{\text{argmin}} \left(f_i(z_i) - y_i^T z_i \right)$$

Dual subgradients

- dual function for $\lambda \geq 0$

$$g(\lambda) = -f^*(-A^T\lambda) - \lambda^T b$$

- subgradient of $-g$ at $\lambda \geq 0$

$$-Ax(y) + b \in \partial(-g)(\lambda)$$

KKT conditions

$$x = x(y), \quad y = -A^T\lambda \qquad (1)$$
$$Ax \leq b \qquad (2)$$
$$\lambda \circ (Ax - b) = 0 \qquad (3)$$
$$\lambda \geq 0 \qquad (4)$$

Optimality condition residuals

$$r_p = \mathbf{1}^T(Ax - b)_+, \qquad r_c = \lambda^T|Ax - b|$$

Multiple-response oracles

agents query *approximate conjugate subgradient oracle* w.r.t. current prices $y \in \mathbf{R}^n$ and return $x^{\text{apx}}(y)$

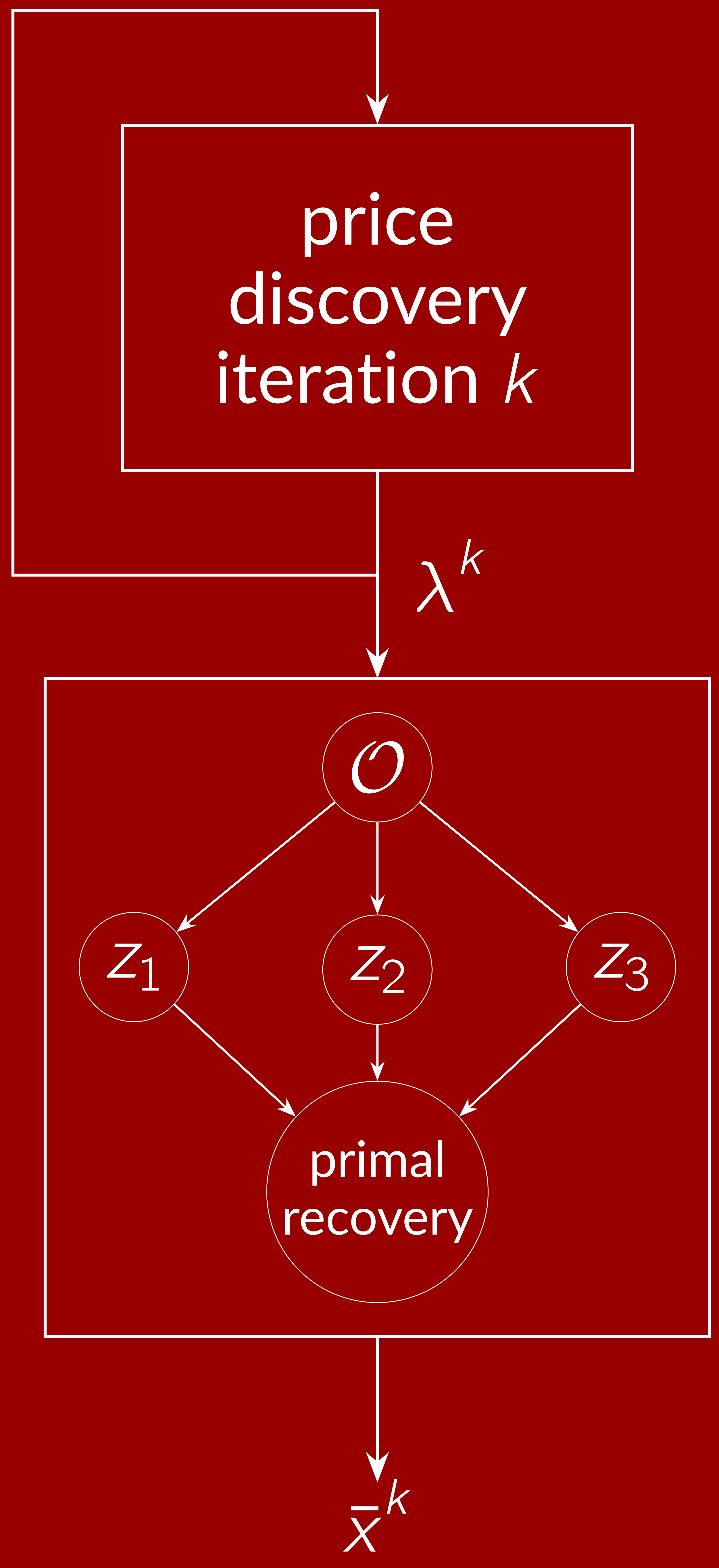
$$-f^*(y) \leq f(x^{\text{apx}}(y)) - y^T x^{\text{apx}}(y) \approx -f^*(y)$$

- value suboptimality oracle

$$-f^*(y) \leq f(x^{\text{v}}(y)) - y^T x^{\text{v}}(y) \leq -f^*(y) + \epsilon_v |f^*(y)|$$

- price perturbation oracle

$$f(x^{\text{p}}(y)) - (y+\delta)^T x^{\text{p}}(y) = -f^*(y+\delta)$$



Primal recovery using MRA

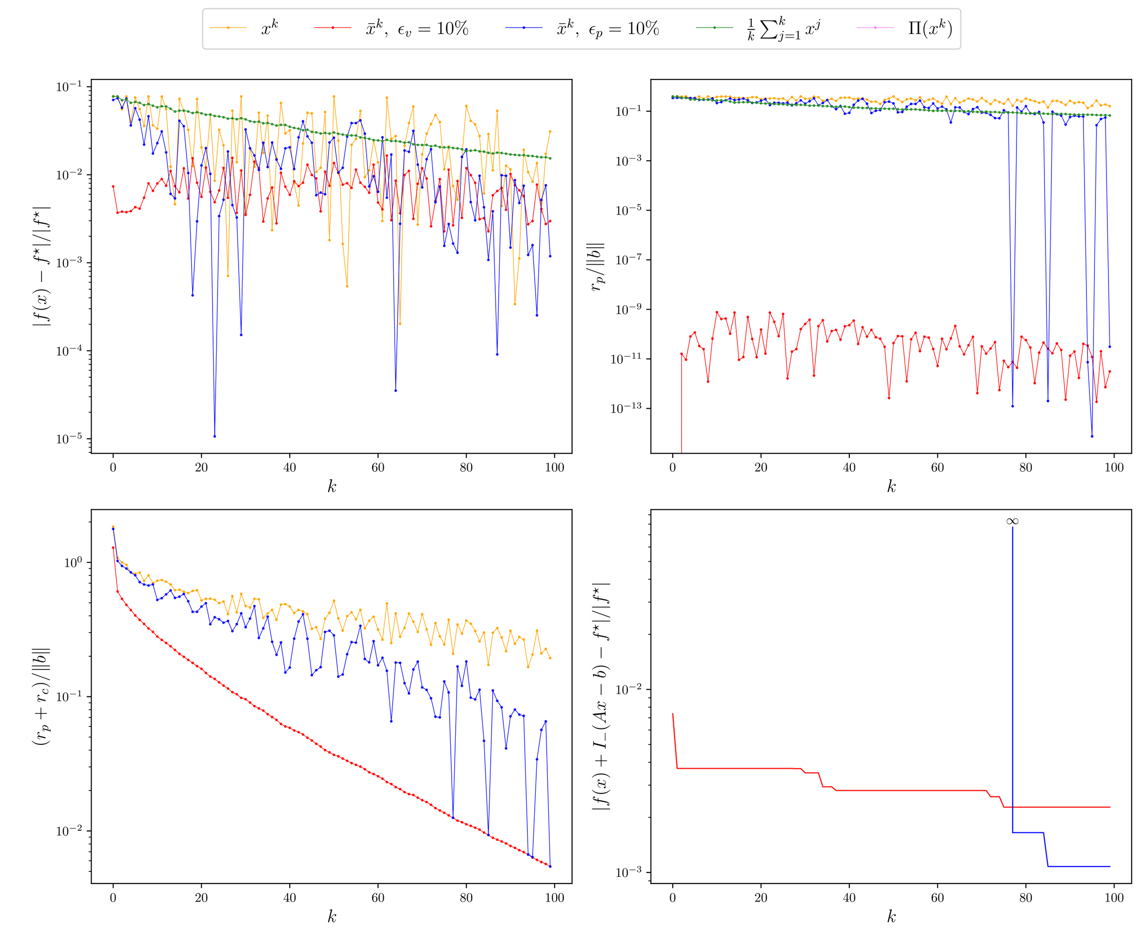
1: approximate oracle queries

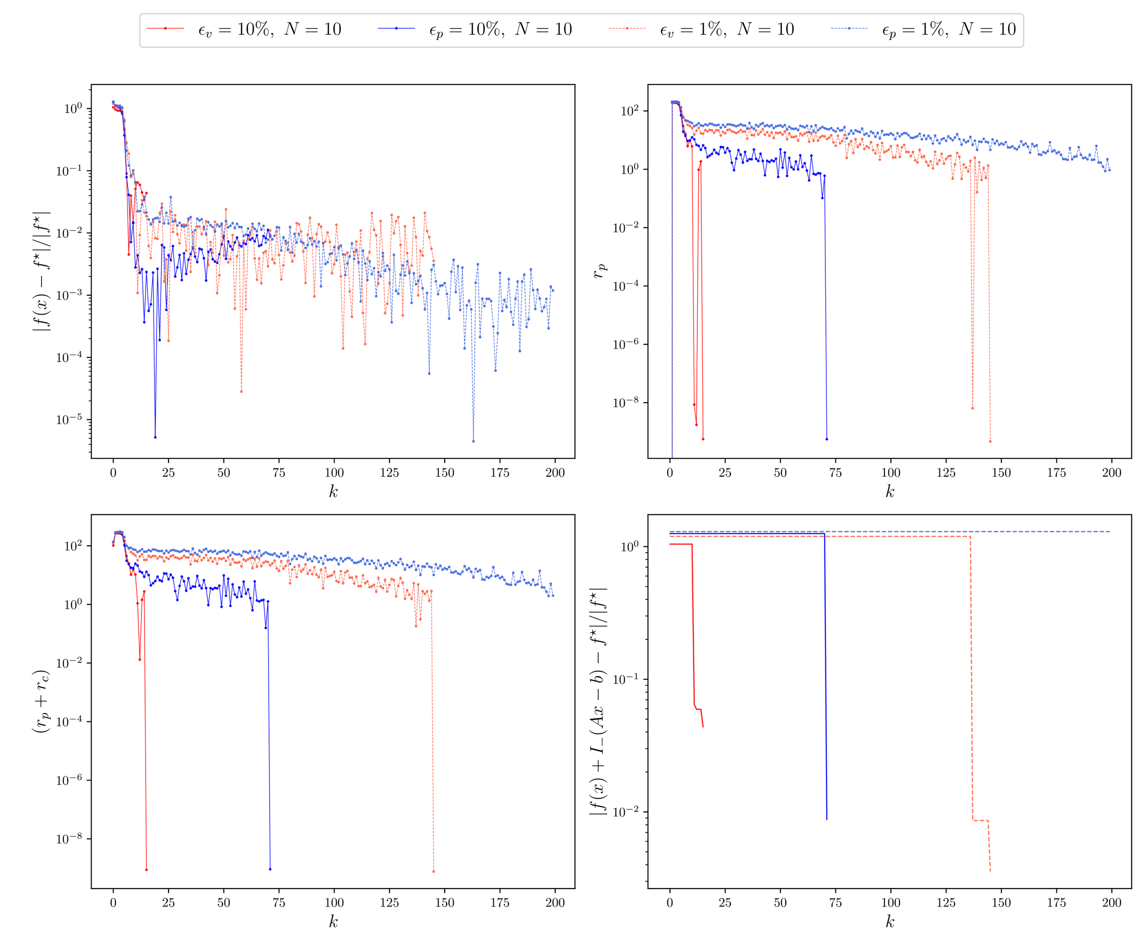
each agent i returns a list of N_i ϵ-suboptimal primal points $Z_i = \{z_{I_1}, \ldots, z_{I_{N_i}}\}$, associated with local price vector $y_i = -A_i^T\lambda$

2: primal recovery of $\bar{x} = (\bar{x}_1, \ldots, \bar{x}_K)$

$$\begin{array}{ll} \text{minimize} & r_p + r_c \\ \text{subject to} & r_p = \mathbf{1}^T(A\bar{x} - b)_+, \qquad r_c = \lambda^T|A\bar{x} - b| \\ & \bar{x}_i = Z_i u_i, \quad i = 1, \ldots, K \\ & \mathbf{1}^T u_i = 1, \quad i = 1, \ldots, K \\ & u_i \geq 0, \quad i = 1, \ldots, K \end{array}$$

Numerical results





Contribution

- new primal recovery approach to generate fast, near-optimal, near-feasible primal solution
- in practice MRA converges to a reasonable approximate solution in just a few tens of iterations
- due to parallel calls, MRA increases the total computation budget but not the wall clock time of the underlying dual algorithm
- hyperparameter tuning for trading speed and solution quality

github.com/cvxgrp/mra_precovery