An Efficient Algorithm for Large-Scale Linear and Convex Minimization in Relative Scale

Peter Richtárik
INMA, Catholic University of Louvain, Belgium

(1) Summary

We develop a single algorithm simultaneously solving five convex optimization problems within a prescribed relative error ϵ in $O(1/\epsilon)$ gradient-type iterations [3]. This is possible due to the presence of central symmetry. Rank-1 updates of a square matrix (i.e. matrix-vector multiplications) are the dominating cost of every iteration and hence the method is suited for large-scale problems.

- In convex optimization, virtually all algorithms until recently [2] aim for an ϵ-solution in absolute scale. This is in contrast with combinatorial optimization where approximation algorithms are studied extensively.
- Until recently, it was believed that the best iteration complexity obtainable for nonsmooth convex problems is $O(1/\epsilon^2)$. While this is true for the first-order black-box methods (the subgradient method is optimal), it was shown [1], [2] that methods exploiting the structure of the problem can bring this down to $O(1/\epsilon)$.
- Our approach reveals connections, appearing under symmetry, between several classes of optimizations algorithms.

(2) Five related optimization problems

Input: Vectors $d; a_1, a_2, \dots, a_m \in \mathbf{R}^n$; $m \gg n$

Assumptions: Vectors a_i span $\mathbf{R}^n$ and $d \neq 0$

Essential notation:
$A = [a_1, \dots, a_m]$ ($n \times m$ matrix)
$\Delta_m = \{w \in \mathbf{R}^m : w_i \geq 0, \sum w_i = 1\}$ (simplex)
$\mathcal{Q} = \mathrm{conv}\{\pm a_i\}$ (a centrally symmetric convex body)
$\mathcal{Q}^0 = \{x : \langle a_i, x\rangle \leq 1 \; \forall i\}$ (the polar of $\mathcal{Q}$)
$U(w) = \sum_i w_i a_i a_i^T = A\,\mathrm{diag}(w) A^T$ (psd for $w \in \mathrm{int}\, \Delta_m$)
$\|d\|^*_{U(w)} = \langle d, U(w)^{-1} d\rangle^{1/2}$

We consider the following problems:

$$(P1) \qquad \varphi^* = \min_x \{\max_i |\langle a_i, x\rangle| : \langle d, x\rangle = 1\}$$

$$(D1) \qquad \varphi^* = \max_\tau \{\tau : \tau d \in \mathcal{Q}\} \qquad \text{DUAL of (P1)}$$

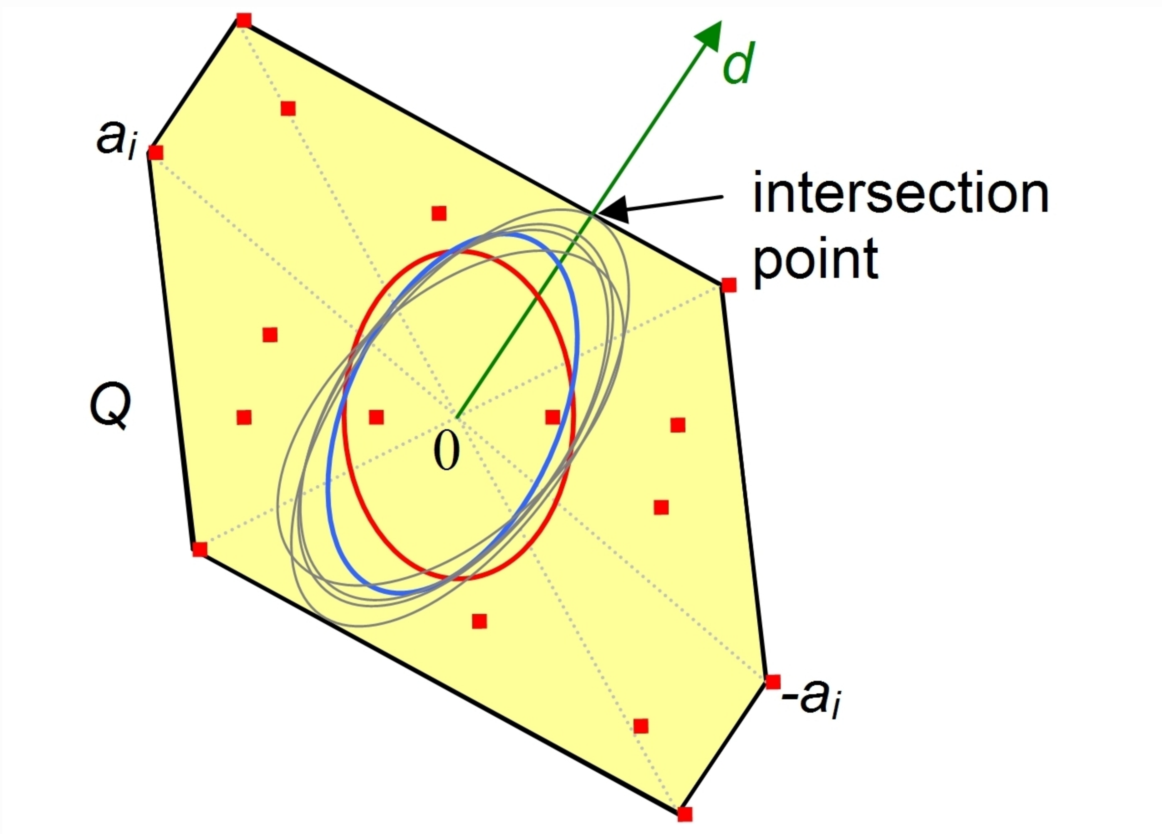
$$(P2) \qquad \tfrac{1}{\varphi^*} = \max_z \{\langle d, z\rangle : z \in \mathcal{Q}^0\}$$

$$(D2) \qquad \tfrac{1}{\varphi^*} = \min_v \{\|v\|_1 : Av = d, \; v \in \mathbf{R}^m\} \qquad \text{DUAL of (P2)}$$

$$(P3) \qquad \tfrac{1}{\varphi^*} = \min_w \{\|d\|^*_{U(w)} : w \in \Delta_m\}$$

Parts (3a)-(3d) of this poster outline the main idea behind the algorithm from the perspective of each of the problems above.

(3a) Intersection of a symmetric convex set with a line



Problems (D1)+(P3): Find the intersection of the centrally symmetric polytope $\mathcal{Q}$ and the ray emanating from the origin in the direction d.

Algorithm: Given an ellipsoid contained within $\mathcal{Q}$ (red ellipsoid on the picture), the next iterate ellipsoid (blue) is obtained by "trying to eat" as large a portion of the ray $\{\tau d : \tau \geq 0\}$ as possible subject to the constraint that it constitutes only a rank-1 update in the psd matrix $U(w)$ defining the ellipsoid and the new ellipsoid is still contained within $\mathcal{Q}$.

Interpretation 1: A modification of Khachiyan's ellipsoidal rounding algorithm.

Interpretation 2: The Frank-Wolfe algorithm on the unit simplex Δ_m applied to $\|d\|^*_{U(\cdot)}$ with explicit line-search.

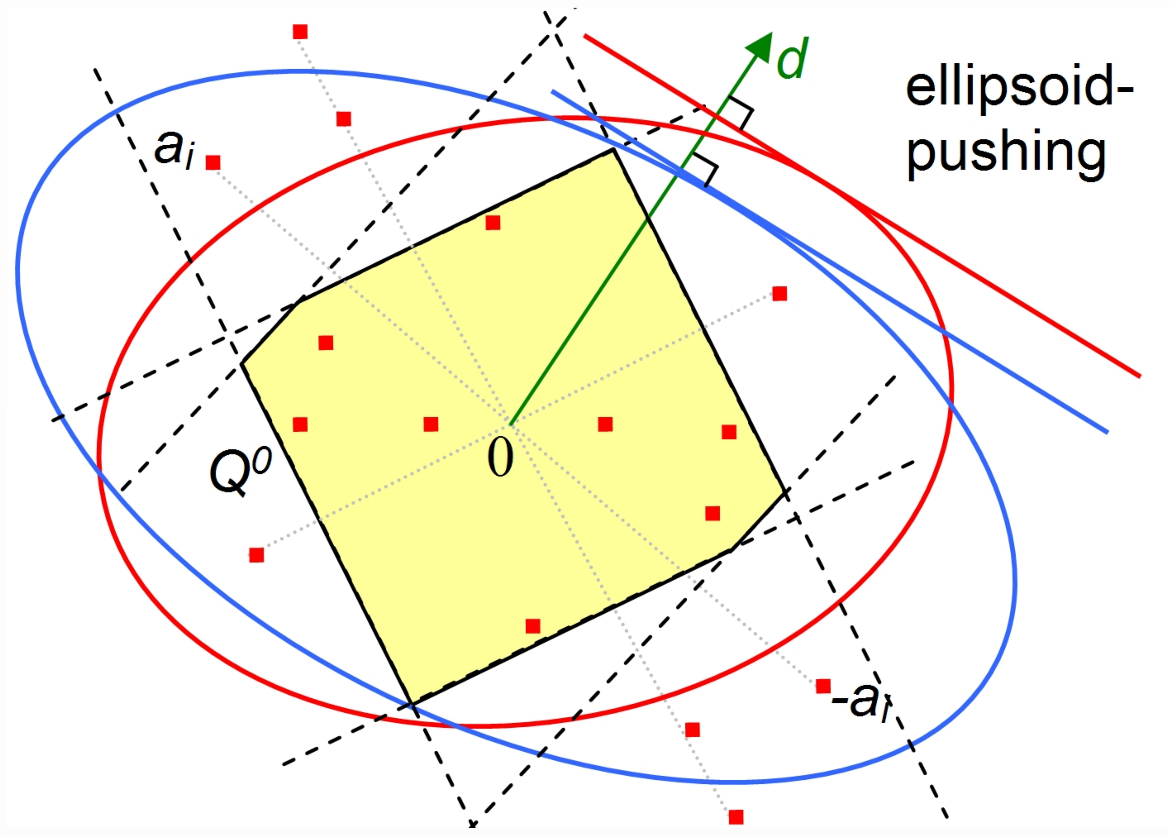
(3b) Unconstrained convex minimization

Problem (P1): Minimize the convex homogeneous function $\varphi(x) := \max_i |\langle a_i, x\rangle|$ on the hyperplane $\{x : \langle d, x\rangle = 1\}$.

Note: All unconstrained convex minimization problems can essentially be written in this form (possibly with $m = \infty$).

Algorithm: If we could replace the objective function φ by a Euclidean norm, then the problem reduces to finding the projection of the origin onto a hyperplane. Instead, we produce a sequence of Euclidean norms (corresponding to positive definite matrices $U(w)$), successively differing by a rank-1 matrix, which approximate the objective function increasingly well near the minimizer. The iterates of the algorithm are the Euclidean projections.

(3c) Symmetric linear programming



Problem (P2): Maximize the linear functional $\langle d, \cdot\rangle$ over the centrally symmetric polytope $\mathcal{Q}^0$.

Algorithm: Given an ellipsoid containing $\mathcal{Q}^0$ (red ellipsoid on the picture), the next iterate ellipsoid (blue) is obtained by "pushing" the previous one greedily "by the hyperplane" with normal d in the direction $-d$ subject to the constraint that it constitutes only a rank-1 update in the psd matrix $U(w)$ defining the ellipsoid and the new ellipsoid still contains $\mathcal{Q}^0$.

Interpretation: A novel ellipsoid-pushing algorithm for symmetric linear programming.

Note: Iterates of this method lie on the boundary of the feasible set.

(3d) ℓ_1 projection onto a subspace

Problem (D2): Find the ℓ_1 projection of the origin onto the subspace $\{v \in R^m : Av = d\}$, or, equivalently, find the least-ℓ_1-norm solution of the full-rank underdetermined linear system $Av = d$.

Algorithm:

- Solve the least-squares problem

$$v = \arg\min\{\|\mathrm{Diag}(w)^{-1/2} v\|_2 : Av = d, v \in \mathbf{R}^m\}$$

- update weights w and iterate.

Interpretation: An iteratively reweighted least squares (IRLS) algorithm where **no** least-squares problem needs to be solved since all the computation can be done in terms of the "dual" vector w.

(4) SAMPLE APPLICATION: Truss topology design

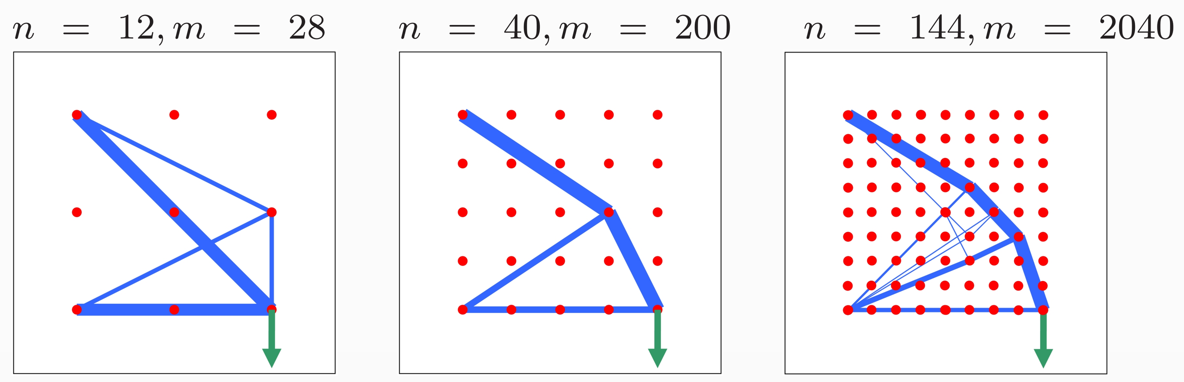
Given:

- a structure of **nodes** and **potential bars** in 2D or 3D of **unit weight**
- a set of **fixed nodes**
- a vector of **forces** applied at the **free nodes**

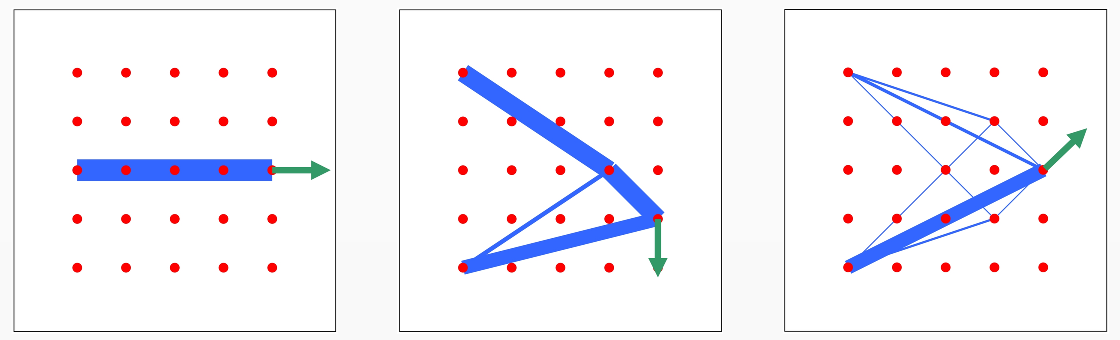
What happens: The structure deforms (nodes move) into an equilibrium position, storing energy.

Goal: minimize the total stiffness (energy stored) of the system.

This problem can be modeled as (P3) with $n = 2\times$# free nodes (or $3\times$), $m =$# potential bars, $w =$ bar weights, $d =$ vector of forces and a_i and $U(w)$ coming out of the equilibrium equation. The plots below are optimal trusses output by our algorithm:



Discretizations of increasing density (3×3, 5×5 and 9×9) and a downward unit force applied at the bottom-right node.



Fixed 5×5 node discretization and unit forces applied at various nodes and angles.

References

[1] Yu. Nesterov, "Smooth minimization of non-smooth functions". Math. Prog., 103(1):127-152, 2005.
[2] Yu. Nesterov, "Unconstrained convex minimization in relative scale". CORE Discussion paper #2003/96.
[3] P. Richtárik, "Some Algorithms for Large-Scale Linear and Convex Minimization in Relative Scale", Ph.D. thesis, Cornell University, 2007.