Blended Conditional Gradients:
The unconditioning of conditional gradients

Joint work with Gabor Braun, Sebastian Pokutta, Steve Wright

Dan Tu

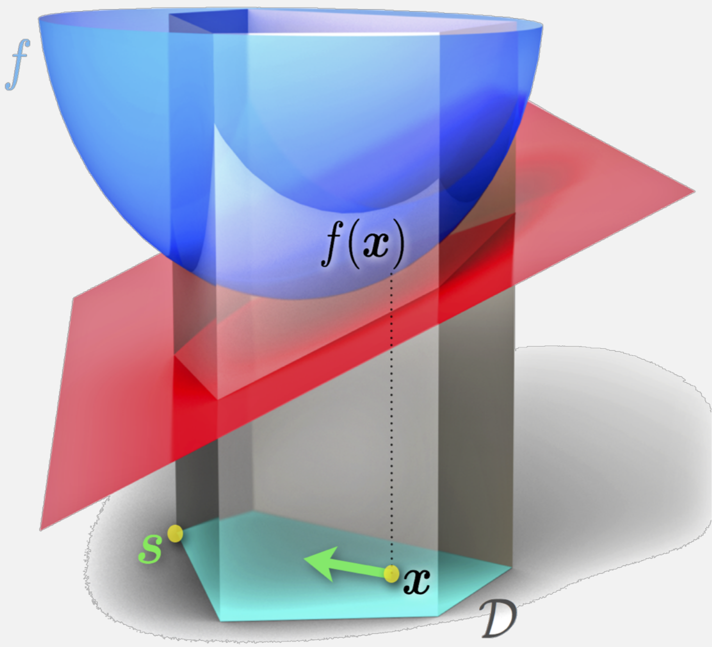
Georgia Tech Machine Learning

CONDITIONAL GRADIENTS: PROJECTION-FREE

Given a polytope P, solve the optimization problem:

$$\min f(x) \qquad \text{s.t. } x \in P$$

where the objective function f is smooth and strongly convex



Frank-Wolfe Algorithm

1: **Input:** initial point $x_0 \in P$, convex function f
2: **for** $t = 1$ to T **do**
3: $s_{t+1} = \text{argmin}_{s \in P} \nabla f(x_t)^\top s$
4: $x_{t+1} = (1-\gamma)x_t + \gamma s_{t+1}$
5: **end for**

Find a vertex through LP Oracle.

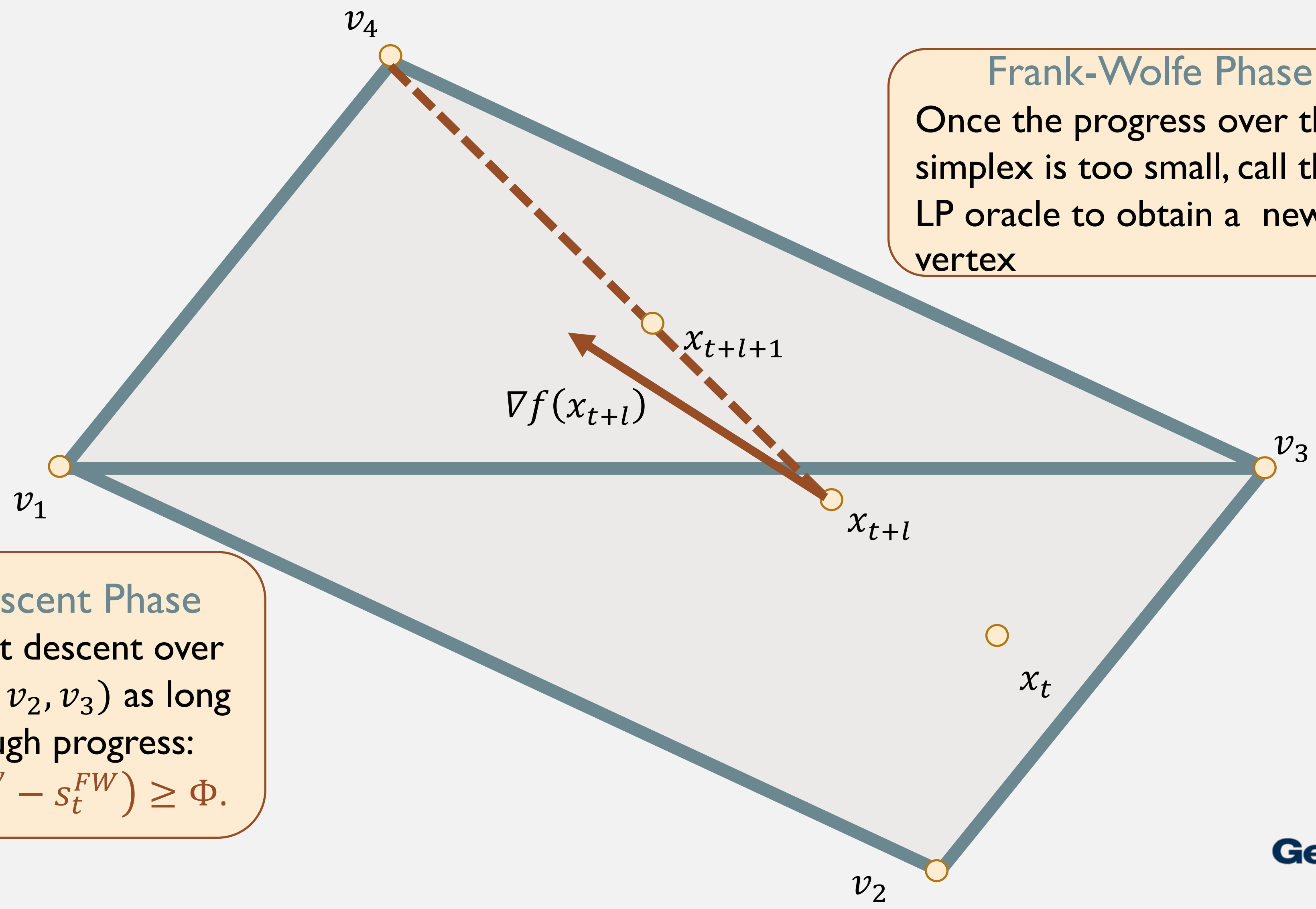
Walk along the conditional gradient direction.

Problems:

- LP Oracle can be computationally expensive.
- The conditional gradient direction, as an approximation of the negative gradient, can be inefficient.

Georgia Tech Machine Learning

BLENDED CONDITIONAL GRADIENT



Frank-Wolfe Phase

Once the progress over the simplex is too small, call the LP oracle to obtain a new vertex

Gradient Descent Phase

Perform gradient descent over the simplex (v_1, v_2, v_3) as long as it makes enough progress: $\nabla f(x_t)^T \left(v_t^{Away} - s_t^{FW}\right) \geq \Phi.$

Georgia Tech Machine Learning

GRADIENT DESCENT PHASE
SIMPLEX GRADIENT DESCENT ORACLE

For a general simplex, decompose x as a convex combination

$$x = \sum_{i=1}^{t} \lambda_i v_i, \text{ with } \sum_{i=1}^{t} \lambda_i = 1 \text{ and } \lambda_i \geq 0, i = 1, 2, \ \dots, \ t$$

Treat λ_i as variables → x in a standard simplex with normal vector:

$$N = (1, 1, \dots, 1)/\sqrt{t}$$

Georgia Tech Machine Learning

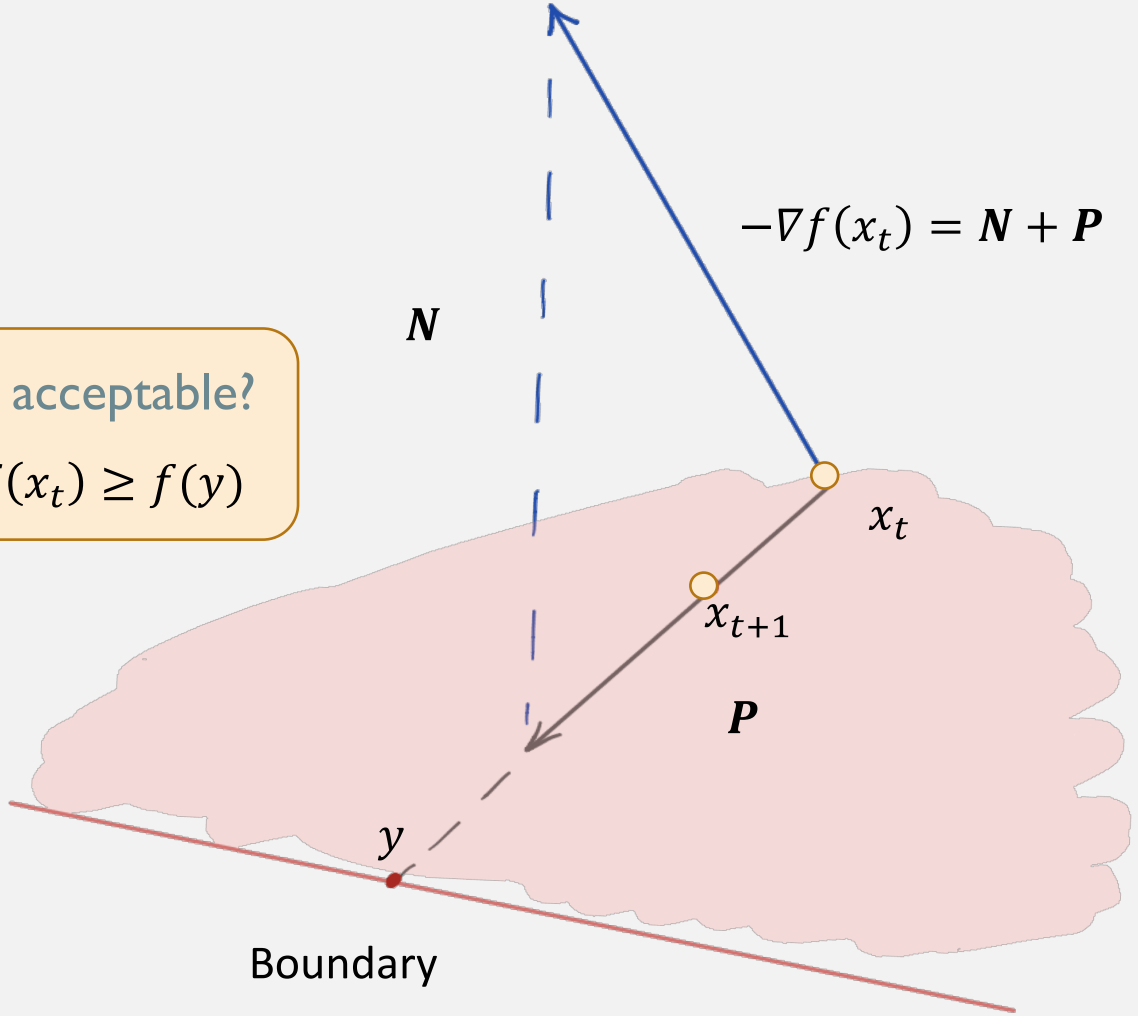
GRADIENT DESCENT PHASE
SIMPLEX GRADIENT DESCENT ORACLE

Decompose $-\nabla f(x_t)$

$$-\nabla f(x_t) = \boldsymbol{N} + \boldsymbol{P}$$
$$\boldsymbol{N} \perp \boldsymbol{P}$$

boundary point acceptable?

Set $x_{t+1} = y$ if $f(x_t) \geq f(y)$



If not acceptable

Perform line search on line segment between x_t and y

BCG ALGORITHM

Algorithm Blended Conditional Gradients (BCG)

Require: smooth convex function f, start vertex $x_0 \in P$, weak separation oracle LPsep_P, accuracy $K \geq 1$

Ensure: points x_t in P for $t = 1, \dots, T$

$\Phi_0 \leftarrow \max_{v \in P} \nabla f(x_0)(x_0 - v)/2$ {Initial dual gap estimate}
$S_0 \leftarrow \{x_0\}$
for $t = 0$ **to** $T - 1$ **do**
  $v_t^A \leftarrow \mathrm{argmax}_{v \in S_t} \nabla f(x_t) v$
  $v_t^{FW-S} \leftarrow \mathrm{argmin}_{v \in S_t} \nabla f(x_t) v$
  if $\nabla f(x_t)(v_t^A - v_t^{FW-S}) \geq \Phi_t$ **then**
    $x_{t+1}, S_{t+1} \leftarrow$ Simplex Gradient Descent(x_t, S_t)
    $\Phi_{t+1} \leftarrow \Phi_t$
  else
    $v_t \leftarrow \mathrm{LPsep}_P(\nabla f(x_t), x_t, \Phi_t, K)$
    if $v_t =$ **false then**
      $x_{t+1} \leftarrow x_t$
      $\Phi_{t+1} \leftarrow \Phi_t/2$ {update dual gap estimate}
      $S_{t+1} \leftarrow S_t$
    else
      $x_{t+1} \leftarrow \mathrm{argmin}_{x \in [x_t, v_t]} f(x)$
      Choose $S_{t+1} \subseteq S_t \cup \{v_t\}$ minimal such that $x_{t+1} \in S_{t+1}$.
      $\Phi_{t+1} \leftarrow \Phi_t$
    end if
  end if
end for

Gradient Descent Phase

Frank-Wolfe Phase

COMPUTATIONAL RESULTS

BCG outperforms several recent variants of Frank-Wolfe algorithm

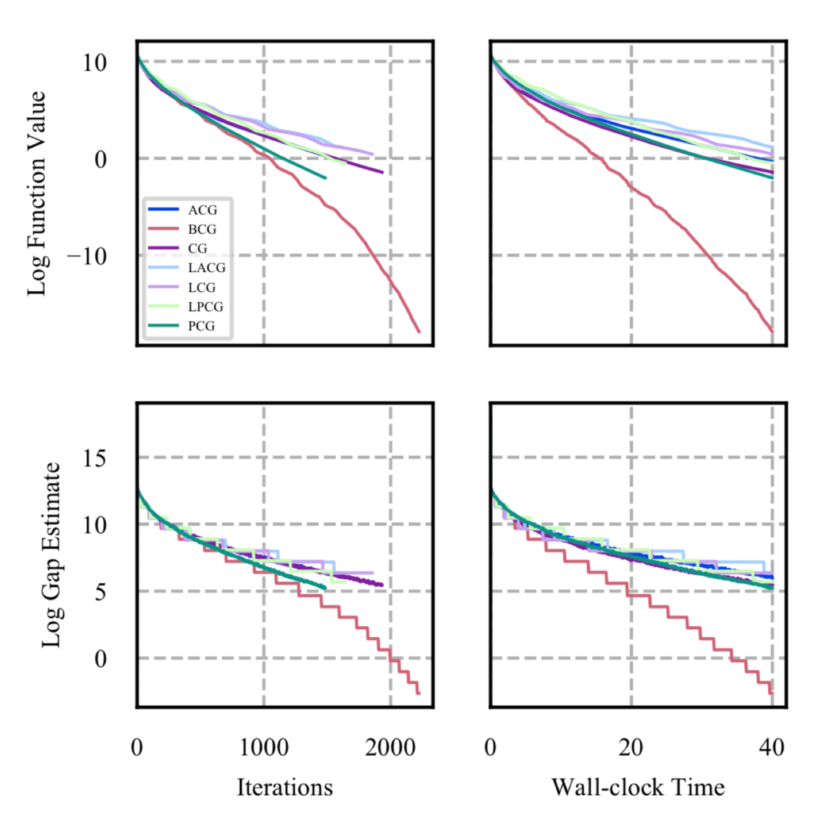


Fig 1: Lasso Regression

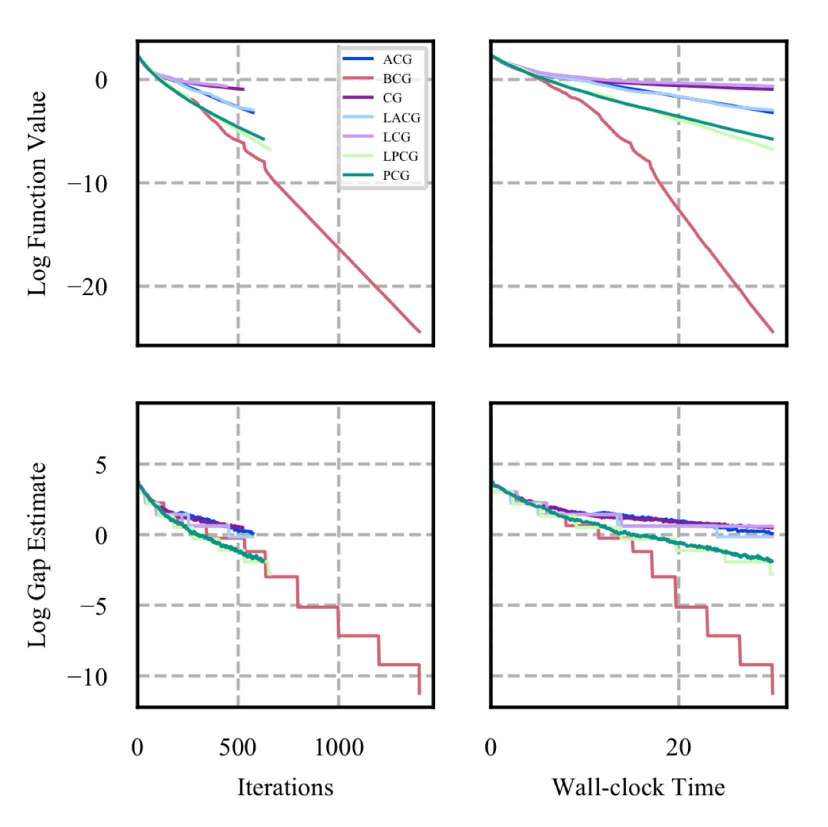


Fig 2: Sparse Signal Recovery

CONVERGENCE

Theorem

If f is a strongly convex and smooth function over the polytope P with geometric strong convexity μ and simplicial curvature, then BCG algorithm ensures $f(x_t) - f(\mathrm{x}^*) \leq \epsilon$ for some T that satisfies:

$$T \leq \left\lceil \log \frac{2\Phi_0}{\epsilon} \right\rceil + 8 \left\lceil \log \frac{\Phi_0}{2C^\Delta} \right\rceil + \frac{64C^\Delta}{\mu} \left\lceil \log \frac{4C^\Delta}{\epsilon} \right\rceil = O\left(\frac{C^\Delta}{\mu} \log \frac{\Phi_0}{\epsilon} \right)$$

Georgia Tech Machine Learning

THANKS!

Georgia Tech Machine Learning