

More First-Order Optimization Algorithms

Yinyu Ye

Department of Management Science and Engineering

Stanford University

Stanford, CA 94305, U.S.A.

<http://www.stanford.edu/~yyye>

Chapters 4.2, 8.4-5, 9.1-7, 12.3-6

Double-Directions: The QP Heavy-Ball Method (Polyak 64)

$$\mathbf{x}^{k+1} = \mathbf{x}^k - \frac{4}{(\sqrt{\lambda_n} + \sqrt{\lambda_1})^2} \nabla f(\mathbf{x}^k) + \left(\frac{\sqrt{\lambda_n} - \sqrt{\lambda_1}}{\sqrt{\lambda_n} + \sqrt{\lambda_1}} \right) (\mathbf{x}^k - \mathbf{x}^{k-1}).$$

where the convergence rate can be improved to

$$\left(\frac{\sqrt{\lambda_n} - \sqrt{\lambda_1}}{\sqrt{\lambda_n} + \sqrt{\lambda_1}} \right)^2.$$

This is also called the **Parallel-Tangent or Conjugate Direction** method, where the second direction-term in the formula is nowadays called “**acceleration**” or “**momentum**” direction.

For minimizing general functions, we can let

$$\mathbf{x}^{k+1} = \mathbf{x}^k - \alpha^g \nabla f(\mathbf{x}^k) + \alpha^m (\mathbf{x}^k - \mathbf{x}^{k-1}) = \mathbf{x}^k + \mathbf{d}(\alpha^g, \alpha^m),$$

where the pair of step-sizes (α^g, α^m) can be chosen to

$$\min_{(\alpha^g, \alpha^d)} \nabla f(\mathbf{x}^k) \mathbf{d}(\alpha^g, \alpha^m) + \frac{1}{2} \mathbf{d}(\alpha^g, \alpha^m) \nabla^2 f(\mathbf{x}^k) \mathbf{d}(\alpha^g, \alpha^m),$$

where \mathbf{x}^1 can be computed from the SDM step.

DRSOM: The Close-Form Step-Size from Newton for Convex Minimization

Let $\mathbf{d}^k = \mathbf{x}^k - \mathbf{x}^{k-1}$, $\mathbf{g}^k = \nabla f(\mathbf{x}^k)$ and $H^k = \nabla^2 f(\mathbf{x}^k)$, then the step-sizes can be chosen from

$$\begin{pmatrix} (\mathbf{g}^k)^T H^k \mathbf{g}^k & -(\mathbf{d}^k)^T H^k \mathbf{g}^k \\ -(\mathbf{d}^k)^T H^k \mathbf{g}^k & (\mathbf{d}^k)^T H^k \mathbf{d}^k \end{pmatrix} \begin{pmatrix} \alpha^g \\ \alpha^m \end{pmatrix} = \begin{pmatrix} \|\mathbf{g}^k\|^2 \\ -(\mathbf{g}^k)^T \mathbf{d}^k \end{pmatrix}.$$

If the Hessian $\nabla^2 f(\mathbf{x}^k)$ is not available, one can approximate

$$H^k \mathbf{g}^k \sim \nabla(\mathbf{x}^k + \mathbf{g}^k) - \mathbf{g}^k \quad \text{and} \quad H^k \mathbf{d}^k \sim \nabla(\mathbf{x}^k + \mathbf{d}^k) - \mathbf{g}^k \sim -(\mathbf{g}^{k-1} - \mathbf{g}^k);$$

or for some small $\epsilon > 0$:

$$H^k \mathbf{g}^k \sim \frac{1}{\epsilon} (\nabla(\mathbf{x}^k + \epsilon \mathbf{g}^k) - \mathbf{g}^k) \quad \text{and} \quad H^k \mathbf{d}^k \sim \frac{1}{\epsilon} (\nabla(\mathbf{x}^k + \epsilon \mathbf{d}^k) - \mathbf{g}^k).$$

“Dimension-Reduced Second-Order Method”: Application in **Federated-Learning**.

The Accelerated Steepest Descent Method (ASDM)

There is an **accelerated** steepest descent method (Nesterov 83) that works as follows:

$$\lambda^0 = 0, \lambda^{k+1} = \frac{1 + \sqrt{1 + 4(\lambda^k)^2}}{2}, \alpha^k = \frac{1 - \lambda^k}{\lambda^{k+1}}, \quad (1)$$

$$\tilde{\mathbf{x}}^{k+1} = \mathbf{x}^k - \frac{1}{\beta} \nabla f(\mathbf{x}^k), \mathbf{x}^{k+1} = (1 - \alpha^k) \tilde{\mathbf{x}}^{k+1} + \alpha^k \tilde{\mathbf{x}}^k. \quad (2)$$

Note that $(\lambda^k)^2 = \lambda^{k+1}(\lambda^{k+1} - 1)$, $\lambda^k > k/2$ and $\alpha^k \leq 0$.

One can prove:

Theorem 1

$$f(\tilde{\mathbf{x}}^{k+1}) - f(\mathbf{x}^*) \leq \frac{2\beta}{k^2} \|\mathbf{x}^0 - \mathbf{x}^*\|^2, \forall k \geq 1.$$

Convergence Analysis of ASDM

Again for simplification, we let $\Delta^k = \lambda^k \mathbf{x}^k - (\lambda^k - 1)\tilde{\mathbf{x}}^k - \mathbf{x}^*$, $\mathbf{g}^k = \nabla f(\mathbf{x}^k)$ and $\delta^k = f(\tilde{\mathbf{x}}^k) - f(\mathbf{x}^*) (\geq 0)$ in the following.

Applying Lemma 1 for $\mathbf{x} = \tilde{\mathbf{x}}^{k+1}$ and $\mathbf{y} = \tilde{\mathbf{x}}^k$, convexity of f and (2) we have

$$\begin{aligned}
 \delta^{k+1} - \delta^k &= f(\tilde{\mathbf{x}}^{k+1}) - f(\mathbf{x}^k) + f(\mathbf{x}^k) - f(\tilde{\mathbf{x}}^k) \\
 &\leq -\frac{\beta}{2} \|\tilde{\mathbf{x}}^{k+1} - \mathbf{x}^k\|^2 + f(\mathbf{x}^k) - f(\tilde{\mathbf{x}}^k) \\
 &\leq -\frac{\beta}{2} \|\tilde{\mathbf{x}}^{k+1} - \mathbf{x}^k\|^2 + (\mathbf{g}^k)^T (\mathbf{x}^k - \tilde{\mathbf{x}}^k) \\
 &= -\frac{\beta}{2} \|\tilde{\mathbf{x}}^{k+1} - \mathbf{x}^k\|^2 - \beta (\tilde{\mathbf{x}}^{k+1} - \mathbf{x}^k)^T (\mathbf{x}^k - \tilde{\mathbf{x}}^k).
 \end{aligned} \tag{3}$$

Applying Lemma 1 for $\mathbf{x} = \tilde{\mathbf{x}}^{k+1}$ and $\mathbf{y} = \mathbf{x}^*$, convexity of f and (2) we have

$$\begin{aligned}
 \delta^{k+1} &= f(\tilde{\mathbf{x}}^{k+1}) - f(\mathbf{x}^k) + f(\mathbf{x}^k) - f(\mathbf{x}^*) \\
 &\leq -\frac{\beta}{2} \|\tilde{\mathbf{x}}^{k+1} - \mathbf{x}^k\|^2 + f(\mathbf{x}^k) - f(\mathbf{x}^*) \\
 &\leq -\frac{\beta}{2} \|\tilde{\mathbf{x}}^{k+1} - \mathbf{x}^k\|^2 + (\mathbf{g}^k)^T (\mathbf{x}^k - \mathbf{x}^*) \\
 &= -\frac{\beta}{2} \|\tilde{\mathbf{x}}^{k+1} - \mathbf{x}^k\|^2 - \beta (\tilde{\mathbf{x}}^{k+1} - \mathbf{x}^k)^T (\mathbf{x}^k - \mathbf{x}^*).
 \end{aligned} \tag{4}$$

Multiplying (3) by $\lambda^k(\lambda^k - 1)$ and (4) by λ^k respectively, and summing the two, we have

$$\begin{aligned}
 (\lambda^k)^2 \delta^{k+1} - (\lambda^{k-1})^2 \delta^k &\leq -(\lambda^k)^2 \frac{\beta}{2} \|\tilde{\mathbf{x}}^{k+1} - \mathbf{x}^k\|^2 - \lambda^k \beta (\tilde{\mathbf{x}}^{k+1} - \mathbf{x}^k)^T \Delta^k \\
 &= -\frac{\beta}{2} ((\lambda^k)^2 \|\tilde{\mathbf{x}}^{k+1} - \mathbf{x}^k\|^2 + 2\lambda^k (\tilde{\mathbf{x}}^{k+1} - \mathbf{x}^k)^T \Delta^k) \\
 &= -\frac{\beta}{2} (\|\lambda^k \tilde{\mathbf{x}}^{k+1} - (\lambda^k - 1)\tilde{\mathbf{x}}^k - \mathbf{x}^*\|^2 - \|\Delta^k\|^2) \\
 &= \frac{\beta}{2} (\|\Delta^k\|^2 - \|\lambda^k \tilde{\mathbf{x}}^{k+1} - (\lambda^k - 1)\tilde{\mathbf{x}}^k - \mathbf{x}^*\|^2).
 \end{aligned}$$

Using (1) and (2) we can derive

$$\lambda^k \tilde{\mathbf{x}}^{k+1} - (\lambda^k - 1)\tilde{\mathbf{x}}^k = \lambda^{k+1} \mathbf{x}^{k+1} - (\lambda^{k+1} - 1)\tilde{\mathbf{x}}^{k+1}.$$

Thus,

$$(\lambda^k)^2 \delta^{k+1} - (\lambda^{k-1})^2 \delta^k \leq \frac{\beta}{2} (\|\Delta^k\|^2 - \|\Delta^{k+1}\|^2). \quad (5)$$

Sum up (5) from 1 to k we have

$$\delta^{k+1} \leq \frac{\beta}{2(\lambda^k)^2} \|\Delta^1\|^2 \leq \frac{2\beta}{k^2} \|\Delta^0\|^2$$

since $\lambda^k \geq k/2$ and $\|\Delta^1\| \leq \|\Delta^0\|$.

First-Order Algorithms for Conic Constrained Optimization (CCO)

Consider the conic nonlinear optimization problem: $\min f(\mathbf{x}) \quad \text{s.t.} \quad \mathbf{x} \in K$.

- Nonnegative Linear Regression: given data $A \in R^{m \times n}$ and $\mathbf{b} \in R^m$

$$\min f(\mathbf{x}) = \frac{1}{2} \|A\mathbf{x} - \mathbf{b}\|^2 \quad \text{s.t.} \quad \mathbf{x} \geq \mathbf{0}; \quad \text{where } \nabla f(\mathbf{x}) = A^T(A\mathbf{x} - \mathbf{b}).$$

- Semidefinite Linear Regression: given data $A_i \in S^n$ for $i = 1, \dots, m$ and $\mathbf{b} \in R^m$

$$\min f(X) = \frac{1}{2} \|\mathcal{A}X - \mathbf{b}\|^2 \quad \text{s.t.} \quad X \succeq \mathbf{0}; \quad \text{where } \nabla f(X) = \mathcal{A}^T(\mathcal{A}X - \mathbf{b}).$$

$$\mathcal{A}X = \begin{pmatrix} A_1 \bullet X \\ \vdots \\ A_m \bullet X \end{pmatrix} \quad \text{and} \quad \mathcal{A}^T \mathbf{y} = \sum_{i=1}^m y_i A_i.$$

Suppose we start from a feasible solution \mathbf{x}^0 or X^0 .

SDM Followed by the Conic-Region-Projection

- $\hat{\mathbf{x}}^{k+1} = \mathbf{x}^k - \frac{1}{\beta} \nabla f(\mathbf{x}^k)$
- $\mathbf{x}^{k+1} = \text{Proj}_K(\hat{\mathbf{x}}^{k+1})$: Solve $\min_{\mathbf{x} \in K} \|\mathbf{x} - \hat{\mathbf{x}}^{k+1}\|^2$.

For examples:

- if $K = \{\mathbf{x} : \mathbf{x} \succeq \mathbf{0}\}$, then

$$\mathbf{x}^{k+1} = \text{Proj}_K(\hat{\mathbf{x}}^{k+1}) = \max\{\mathbf{0}, \hat{\mathbf{x}}^{k+1}\}.$$

- If $K = \{X : X \succeq \mathbf{0}\}$, then factorize $\hat{X}^{k+1} = \sum_{j=1}^n \lambda_j \mathbf{v}_j \mathbf{v}_j^T$ and let

$$X^{k+1} = \text{Proj}_K(\hat{X}^{k+1}) = \sum_{j: \lambda_j > 0} \lambda_j \mathbf{v}_j \mathbf{v}_j^T.$$

(The drawback is that the total eigenvalue-factorization may be costly...)

Does the method converge? What is the convergence speed? See more details in HW3.

SDM Followed by the Convex-Region-Projection

Consider the convex-region-constrained nonlinear optimization problem: $\min f(\mathbf{x})$ s.t. $A\mathbf{x} = \mathbf{b}$. that is $K = \{\mathbf{x} : A\mathbf{x} = \mathbf{b}\}$.

The projection method becomes, starting from a feasible solution \mathbf{x}^0 and let direction

$$\mathbf{d}^k = -(I - A^T(AA^T)^{-1}A)\nabla f(\mathbf{x}^k)$$

$$\mathbf{x}^{k+1} = \mathbf{x}^k + \alpha^k \mathbf{d}^k; \tag{6}$$

where the stepsize can be chosen from line-search or again simply let

$$\alpha^k = \frac{1}{\beta}$$

and β is the (global) Lipschitz constant.

Does the method converge? What is the convergence speed? See more details in HW3.

SDM Followed by the Nonconvex-Region-Projection

- $K \subset R^n$ whose support size is no more than $d(< n)$: $\mathbf{x} = \text{Proj}_K(\hat{\mathbf{x}})$ contains the largest d absolute entries of $\hat{\mathbf{x}}$ and set the rest of them to zeros.
- $K \subset R_+^n$ and its support size is no more than $d(< n)$: $\mathbf{x} = \text{Proj}_K(\hat{\mathbf{x}})$ contains the largest no more than d positive entries of $\hat{\mathbf{x}}$ and set the rest of them to zeros.
- $K \subset S^n$ whose rank is no more than $d(< n)$: factorize $\hat{X} = \sum_{j=1}^n \lambda_j \mathbf{v}_j \mathbf{v}_j^T$ with $|\lambda_1| \geq |\lambda_2| \geq \dots \geq |\lambda_n|$ then $\text{Proj}_K(\hat{X}) = \sum_{j=1}^d \lambda_j \mathbf{v}_j \mathbf{v}_j^T$.
- $K \subset S_+^n$ whose rank is no more than $d(< n)$: factorize $\hat{X} = \sum_{j=1}^n \lambda_j \mathbf{v}_j \mathbf{v}_j^T$ with $\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_n$ then $\text{Proj}_K(\hat{X}) = \sum_{j=1}^d \max\{0, \lambda_j\} \mathbf{v}_j \mathbf{v}_j^T$.

Does the method converge? What is the convergence speed? What if $f(\cdot)$ is not a convex function?

Multiplicative-Update I: “Mirror” SDM for CCO

At the k th iterate with $\mathbf{x}^k > \mathbf{0}$:

$$\mathbf{x}^{k+1} = \mathbf{x}^k \cdot \exp\left(-\frac{1}{\beta} \nabla f(\mathbf{x}^k)\right)$$

Note that \mathbf{x}^{k+1} remains positive in the updating process.

The classical Projected SDM update can be viewed as

$$\mathbf{x}^{k+1} = \arg \min_{\mathbf{x} \geq \mathbf{0}} \nabla f(\mathbf{x}^k)^T \mathbf{x} + \frac{\beta}{2} \|\mathbf{x} - \mathbf{x}^k\|^2.$$

One can choose any strongly convex function $h(\cdot)$ and define

$$\mathcal{D}_h(\mathbf{x}, \mathbf{y}) = h(\mathbf{x}) - h(\mathbf{y}) - \nabla h(\mathbf{y})^T (\mathbf{x} - \mathbf{y})$$

and define the update as

$$\mathbf{x}^{k+1} = \arg \min_{\mathbf{x} \geq \mathbf{0}} \nabla f(\mathbf{x}^k)^T \mathbf{x} + \beta \mathcal{D}_h(\mathbf{x}, \mathbf{x}^k).$$

The update above is the result of choosing (negative) **entropy function** $h(\mathbf{x}) = \sum_j x_j \log(x_j)$.

Multiplicative-Update II: Affine Scaling SDM for CCO

At the k th iterate with $\mathbf{x}^k > \mathbf{0}$, let D^k be a diagonal matrix such that

$$D_{jj}^k = x_j^k, \forall j$$

and

$$\mathbf{x}^{k+1} = \arg \min_{\mathbf{x} \geq \mathbf{0}} \nabla f(\mathbf{x}^k)^T \mathbf{x} + \frac{\beta}{2} \|(D^k)^{-1}(\mathbf{x} - \mathbf{x}^k)\|^2,$$

or

$$\mathbf{x}^{k+1} = \mathbf{x}^k - \alpha_k (D^k)^2 \nabla f(\mathbf{x}^k) = \mathbf{x}^k \cdot * (\mathbf{e} - \alpha_k \nabla f(\mathbf{x}^k) \cdot * \mathbf{x}^k)$$

where variable step-sizes can be

$$\alpha^k = \min \left\{ \frac{1}{\beta \max(\mathbf{x}^k)^2}, \frac{1}{2 \|\mathbf{x}^k \cdot * \nabla f(\mathbf{x}^k)\|_\infty} \right\}.$$

Is $\mathbf{x}^k > \mathbf{0}$, $\forall k$? Does it converge? What is the convergence speed? See more details in HW3.

Geometric Interpretation: inscribed **ball** vs inscribed **ellipsoid**.

Affine Scaling for SDP Cone?

At the k th iterate with $X^k \succ \mathbf{0}$, the new SDM iterate would be

$$X^{k+1} = X^k - \alpha_k X^k \nabla f(X^k) X^k = X^k (I - \alpha_k \nabla f(X^k) X^k).$$

Choose step-size is chosen such that the smallest eigenvalue of X^{k+1} is at most a fraction from the one of X^k ?

Does it converge? What is the convergence speed? See more details in HW3.

Reduced Gradient Method – the Simplex Algorithm for LP

$$\text{LP: } \min \mathbf{c}^T \mathbf{x} \quad \text{s.t. } A\mathbf{x} = \mathbf{b}, \mathbf{x} \geq \mathbf{0},$$

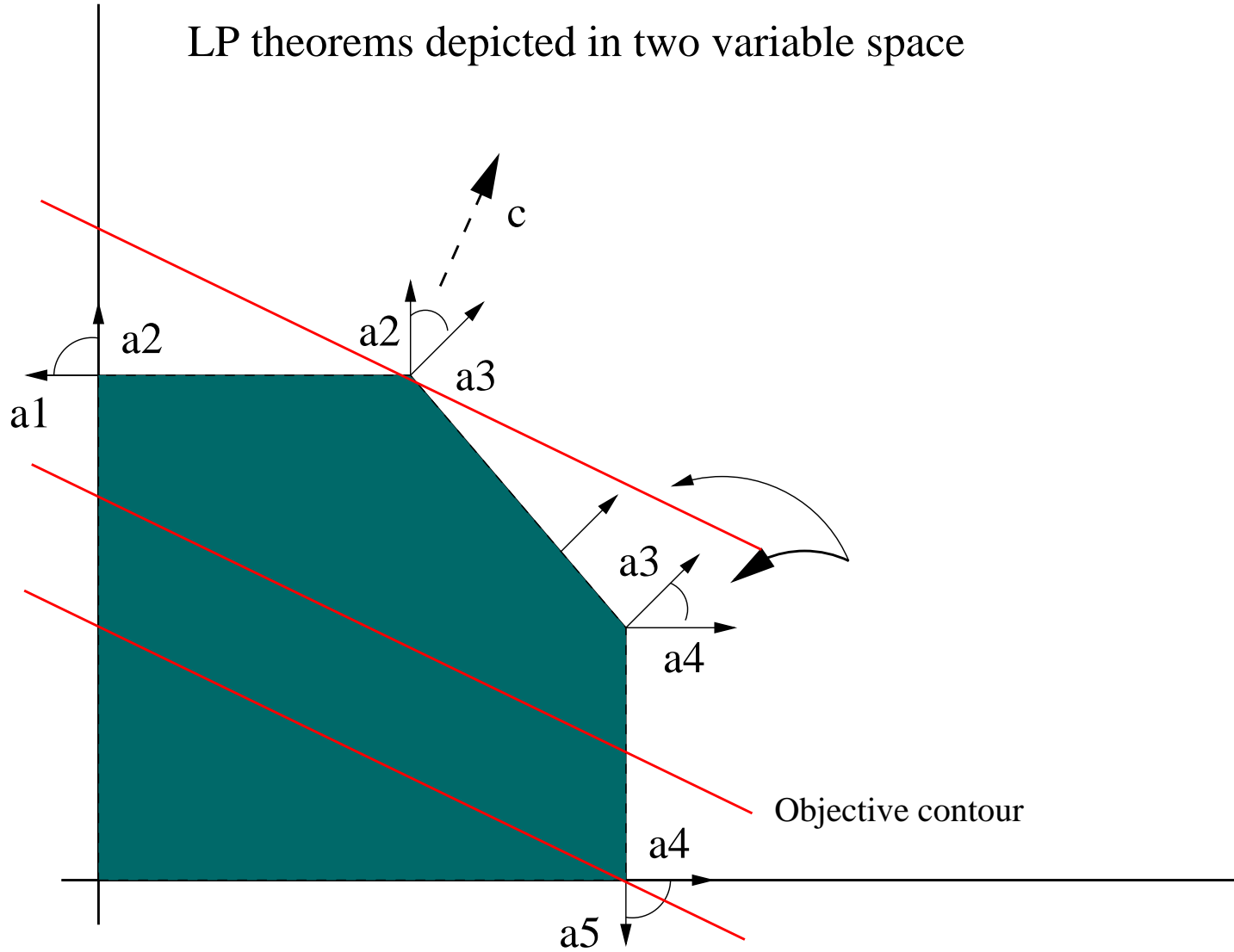
where $A \in \mathbb{R}^{m \times n}$ has a full row rank m .

Theorem 2 (The Fundamental Theorem of LP in Algebraic form) Given (LP) and (LD) where A has full row rank m ,

- i) if there is a feasible solution, there is a *basic feasible solution* (Carathéodory's theorem);
- ii) if there is an optimal solution, there is an *optimal basic solution*.

High-Level Idea:

1. **Initialization** Start at a BSF or corner point of the feasible polyhedron.
2. **Test for Optimality.** Compute the reduced gradient vector at the corner. If no *descent and feasible direction* can be found, stop and claim optimality at the current corner point; otherwise, select a new corner point and go to Step 2.



When a Basic Feasible Solution is Optimal

Suppose the basis of a basic feasible solution is A_B and the rest is A_N . One can transform the equality constraint to

$$A_B^{-1} A \mathbf{x} = A_B^{-1} \mathbf{b}, \quad \text{so that } \mathbf{x}_B = A_B^{-1} \mathbf{b} - A_B^{-1} A_N \mathbf{x}_N.$$

That is, we express \mathbf{x}_B in terms of \mathbf{x}_N , the non-basic variables are active for constraints $\mathbf{x} \geq \mathbf{0}$. Then the objective function equivalently becomes

$$\begin{aligned} \mathbf{c}^T \mathbf{x} &= \mathbf{c}_B^T \mathbf{x}_B + \mathbf{c}_N^T \mathbf{x}_N = \mathbf{c}_B^T A_B^{-1} \mathbf{b} - \mathbf{c}_B^T A_B^{-1} A_N \mathbf{x}_N + \mathbf{c}_N^T \mathbf{x}_N \\ &= \mathbf{c}_B^T A_B^{-1} \mathbf{b} + (\mathbf{c}_N^T - \mathbf{c}_B^T A_B^{-1} A_N) \mathbf{x}_N. \end{aligned}$$

Vector $\mathbf{r}^T = \mathbf{c}^T - \mathbf{c}_B^T A_B^{-1} A$ is called the Reduced Gradient/Cost Vector where $\mathbf{r}_B = \mathbf{0}$ always.

Theorem 3 If Reduced Gradient Vector $\mathbf{r}^T = \mathbf{c}^T - \mathbf{c}_B^T A_B^{-1} A \geq \mathbf{0}$, then the BFS is optimal.

Proof: Let $\mathbf{y}^T = \mathbf{c}_B^T A_B^{-1}$ (called Shadow Price Vector), then \mathbf{y} is a dual feasible solution ($\mathbf{r} = \mathbf{c} - A^T \mathbf{y} \geq \mathbf{0}$) and $\mathbf{c}^T \mathbf{x} = \mathbf{c}_B^T \mathbf{x}_B = \mathbf{c}_B^T A_B^{-1} \mathbf{b} = \mathbf{y}^T \mathbf{b}$, that is, the duality gap is zero.

The Simplex Algorithm Procedures

0. **Initialize** Start a BFS with basic index set B and let N denote the complementary index set.

1. **Test for Optimality:** Compute the **Reduced Gradient Vector** \mathbf{r} at the current BFS and let

$$r_e = \min_{j \in N} \{r_j\}.$$

If $r_e \geq 0$, stop – the current BFS is **optimal**.

2. **Determine the Replacement:** Increase x_e while keep all other non-basic variables at the zero value (inactive) and maintain the equality constraints:

$$\mathbf{x}_B = A_B^{-1} \mathbf{b} - A_B^{-1} A_{.e} x_e \ (\geq \mathbf{0}).$$

If x_e can be increased to ∞ , stop – the problem is **unbounded** below. Otherwise, let the basic variable x_o be the one first becoming 0.

3. **Update basis:** update B with x_o being replaced by x_e , and return to Step 1.

A Toy Example

$$\begin{array}{llllll}
 \text{minimize} & -x_1 & -2x_2 & & & \\
 \text{subject to} & x_1 & & +x_3 & & = 1 \\
 & & x_2 & & +x_4 & = 1 \\
 & & & x_1 & +x_2 & & +x_5 = 1.5.
 \end{array}$$

$$A = \begin{pmatrix} 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 & 1 \end{pmatrix}, \mathbf{b} = \begin{pmatrix} 1 \\ 1 \\ 1.5 \end{pmatrix}, \mathbf{c}^T = (-1 \ -2 \ 0 \ 0 \ 0).$$

Consider initial BFS with basic variables $B = \{3, 4, 5\}$ and $N = \{1, 2\}$.

Iteration 1:

1. $A_B = I$, $A_B^{-1} = I$, $\mathbf{y}^T = (0 \ 0 \ 0)$ and $\mathbf{r}_N = (-1 \ -2)$ – it's **NOT optimal**. Let $e = 2$.

2. Increase x_2 while

$$\mathbf{x}_B = A_B^{-1} \mathbf{b} - A_B^{-1} A_{.2} x_2 = \begin{pmatrix} 1 \\ 1 \\ 1.5 \end{pmatrix} - \begin{pmatrix} 0 \\ 1 \\ 1 \end{pmatrix} x_2.$$

We see x_4 becomes 0 first.

3. The new basic variables are $B = \{3, 2, 5\}$ and $N = \{1, 4\}$.

Iteration 2:

1.

$$A_B = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 1 & 1 \end{pmatrix}, \quad A_B^{-1} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & -1 & 1 \end{pmatrix},$$

$\mathbf{y}^T = (0 \ -2 \ 0)$ and $\mathbf{r}_N = (-1 \ 2)$ – it's **NOT optimal**. Let $e = 1$.

2. Increase x_1 while

$$\mathbf{x}_B = A_B^{-1} \mathbf{b} - A_B^{-1} A_{.1} x_1 = \begin{pmatrix} 1 \\ 1 \\ 0.5 \end{pmatrix} - \begin{pmatrix} 1 \\ 0 \\ 1 \end{pmatrix} x_1.$$

We see x_5 becomes 0 first.

3. The new basic variables are $B = \{3, 2, 1\}$ and $N = \{4, 5\}$.

Iteration 3:

1.

$$A_B = \begin{pmatrix} 1 & 0 & 1 \\ 0 & 1 & 0 \\ 0 & 1 & 1 \end{pmatrix}, \quad A_B^{-1} = \begin{pmatrix} 1 & 1 & -1 \\ 0 & 1 & 0 \\ 0 & -1 & 1 \end{pmatrix},$$

$\mathbf{y}^T = (0 \quad -1 \quad -1)$ and $\mathbf{r}_N = (1 \quad 1)$ – it's **Optimal**.

Is the Simplex Method always convergent to a minimizer? Which condition of the Global Convergence Theorem failed?

The Frank-Wolf Algorithm

$$\text{P: } \min f(\mathbf{x}) \quad \text{s.t. } A\mathbf{x} = \mathbf{b}, \mathbf{x} \geq \mathbf{0},$$

where $A \in \mathbb{R}^{m \times n}$ has a full row rank m .

Start with a feasible solution \mathbf{x}^0 , and at the k th iterate do:

- Solve the LP problem

$$\min \nabla f(\mathbf{x}^k)^T \mathbf{x} \quad \text{s.t. } A\mathbf{x} = \mathbf{b}, \mathbf{x} \geq \mathbf{0}$$

and let $\tilde{\mathbf{x}}^{k+1}$ be an optimal solution.

- Choose a step-size $0 < \alpha^k \leq 1$ and let

$$\mathbf{x}^{k+1} = \mathbf{x}^k + \alpha^k (\tilde{\mathbf{x}}^{k+1} - \mathbf{x}^k).$$

This is also called **sequential linear programming (SLP)** method.

Value-Iteration for MDP I: Fixed-Point Mapping

Let $\mathbf{y} \in \mathbb{R}^m$ represent the **cost-to-go** values of the m states, i th entry for i th state, of a given policy. The MDP problem entails choosing the optimal value vector \mathbf{y}^* which is a fixed-point of:

$$y_i^* = \min_{j \in \mathcal{A}_i} \{c_j + \gamma \mathbf{p}_j^T \mathbf{y}^*\}, \quad \forall i,$$

The Value-Iteration (VI) Method is, starting from any \mathbf{y}^0 , the iterative mapping:

$$y_i^{k+1} = A(\mathbf{y}^k)_j = \min_{j \in \mathcal{A}_i} \{c_j + \gamma \mathbf{p}_j^T \mathbf{y}^k\}, \quad \forall i.$$

If the initial \mathbf{y}^0 is strictly feasible for state i , that is, $y_i^0 < c_j + \gamma \mathbf{p}_j^T \mathbf{y}^0$, $\forall j \in \mathcal{A}_i$, then y_i^k would be increasing in the VI iteration for all i and k .

On the other hand, if any of the inequalities is violated, then we have to decrease y_i^1 at least to

$$\min_{j \in \mathcal{A}_i} \{c_j + \gamma \mathbf{p}_j^T \mathbf{y}^0\}$$

Convergence of Value-Iteration for MDP

Theorem 4 Let the VI algorithm mapping be $A(\mathbf{v})_i = \min_{j \in \mathcal{A}_i} \{c_j + \gamma \mathbf{p}_j^T \mathbf{v}, \forall i\}$. Then, for any two value vectors $\mathbf{u} \in R^m$ and $\mathbf{v} \in R^m$ and every state i :

$$|A(\mathbf{u})_i - A(\mathbf{v})_i| \leq \gamma \|\mathbf{u} - \mathbf{v}\|_\infty, \text{ which implies } \|A(\mathbf{u})_i - A(\mathbf{v})_i\|_\infty \leq \gamma \|\mathbf{u} - \mathbf{v}\|_\infty$$

Let j_u and j_v be the two $\arg \min$ actions for value vectors \mathbf{u} and \mathbf{v} , respectively. Assume that $A(\mathbf{u})_i - A(\mathbf{v})_i \geq 0$ where the other case can be proved similarly.

$$\begin{aligned} 0 \leq A(\mathbf{u})_i - A(\mathbf{v})_i &= (c_{j_u} + \gamma \mathbf{p}_{j_u}^T \mathbf{u}) - (c_{j_v} + \gamma \mathbf{p}_{j_v}^T \mathbf{v}) \\ &\leq (c_{j_v} + \gamma \mathbf{p}_{j_v}^T \mathbf{u}) - (c_{j_v} + \gamma \mathbf{p}_{j_v}^T \mathbf{v}) \\ &= \gamma \mathbf{p}_{j_v}^T (\mathbf{u} - \mathbf{v}) \leq \gamma \|\mathbf{u} - \mathbf{v}\|_\infty. \end{aligned}$$

where the first inequality is from that j_u is the $\arg \min$ action for value vector \mathbf{u} , and the last inequality follows from the fact that the elements in \mathbf{p}_{j_v} are non-negative and sum-up to 1.

Value-Iteration for MDP II: Other issues

The Value-Iteration (VI) Method for zero-sum game, starting from any \mathbf{y}^0 , the iterative mapping:

$$y_i^{k+1} = A(\mathbf{y}^k)_j = \min_{j \in \mathcal{A}_i} \{c_j + \gamma \mathbf{p}_j^T \mathbf{y}^k\}, \forall i \in I^-$$

and

$$y_i^{k+1} = A(\mathbf{y}^k)_j = \max_{j \in \mathcal{A}_i} \{c_j + \gamma \mathbf{p}_j^T \mathbf{y}^k\}, \forall i \in I^+.$$

Remarks':

- One can choose i at random to update, e.g., follow a random walk.
- Aggregate states if they have similar cost-to-go values
- State-values are updated in a **unsynchronized** manner: a state is updated after one of its neighbor-states is updated.

Many research issues in suggested Project III.

Summary of the First-Order Methods

- Good global convergence property (e.g. starting from any (feasible) solution under mild technical assumption...).
- Simple to implement and the computation cost is mainly compute the numerical gradient.
- Maybe difficult to decide step-size: simple back-track is popular in practice.
- The convergence speed can be slow: not suitable for high accuracy computation, certain accelerations available.
- Can only guarantee converging to a first-order KKT solution.