Increasing the Computational Effectiveness of the Simplex Method

February 5, 2004

1 Motivation

SLIDE 1

Many computational enhancements of the simplex method:

- Pre-processing heuristics
- Sparse matrix algebra
- Solving sparse systems of equations
- Setting up Phase I artificial columns and objective function
- Handling variable lower and upper bounds: $l_j \leq x_j \leq u_j$

SLIDE 2

- Handling "range" constraints: $b_i \leq a_i^T x \leq b_i + r_i$
- Working with the basis inverse over a sequence of iterations
- Handling Degeneracy
- Rules for choosing incoming column
- Many others

2 Outline

SLIDE 3

1. Review of the Simplex Algorithm
2. Computation and Matrix Sparsity in the Simplex Algorithm
3. The Simplex Algorithm with Lower and Upper Bounds
4. Working with the Basis Inverse over a Sequence of Iterations

3 Linear Optimization

3.1 General Form

SLIDE 4

$$
\begin{array}{rrcl}
\text{minimize or maximize} & z = c_1 x_1 + \cdots + c_n x_n & & \\
\text{s.t.} & a_{11} x_1 + \cdots + a_{1n} x_n & \leq & b_1 \\
 & & \vdots & \\
 & & \geq & \\
 & & \vdots & \\
 & & = & \\
 & & \vdots & \\
 & a_{m1} x_1 + \cdots + a_{mn} x_n & \vdots & b_m \\
 & x_1, \ldots, x_n & \multicolumn{2}{l}{\geq 0, \leq 0, \text{or free}}
\end{array}
$$

x_j is "free" if x_j has no upper or lower limits.

3.2 Standard Form

SLIDE 5

$$\begin{array}{lllll} \text{minimize} \; z = & c_1 x_1 & + \cdots & + c_n x_n & \\ \text{s.t.} & a_{11} x_1 & + \cdots & + a_{1n} x_n & = \; b_1 \\ & & \vdots & & \vdots \;\; \vdots \\ & a_{m1} x_1 & + \cdots + & a_{mn} x_n & = \; b_m \\ & x_1 & , \; \ldots \; , & x_n & \geq \; 0 \end{array}$$

Convert to matrix notation:

SLIDE 6

$$\begin{array}{lrcl} \text{minimize} \; z = & c^T x & & \\ \text{s.t.} & Ax & = & b \\ & x & \geq & 0 \; . \end{array}$$

$$\begin{array}{lrcl} \text{minimize} \; z = & c^T x & & \\ \text{s.t.} & Ax & = & b \\ & x & \geq & 0 \; . \end{array}$$

We can always conveniently convert any linear optimization model to standard form.

3.3 Example

SLIDE 7

$$\begin{array}{lrcl} \text{minimize} \; z = & c^T x & & \\ \text{s.t.} & Ax & = & b \\ & x & \geq & 0 \; . \end{array}$$

$$c^T = \begin{bmatrix} -1 & 1 & 27 & 5 & 17 & 10 & 16 \end{bmatrix}$$

$$A = \begin{bmatrix} -1 & 1 & 5 & 1 & 2 & 5 & 1 \\ 1 & 0 & 1 & 1 & 1 & 0 & 1 \\ 1 & 1 & 3 & 2 & 0 & 3 & 1 \end{bmatrix} \qquad b = \begin{bmatrix} 4 \\ 3 \\ 9 \end{bmatrix}$$

3.3.1 Initial Tableau

SLIDE 8

Initial Tableau

RHS	x_1	x_2	x_3	x_4	x_5	x_6	x_7
0	−1	1	27	5	17	10	16
4	−1	1	5	1	2	5	1
3	1	0	1	1	1	0	1
9	1	1	3	2	0	3	1

3.3.2 Current Tableau

SLIDE 9

Initial Tableau:

RHS	x_1	x_2	x_3	x_4	x_5	x_6	x_7
0	-1	1	27	5	17	10	16
4	-1	1	5	1	2	5	1
3	1	0	1	1	1	0	1
9	1	1	3	2	0	3	1

After several iterations of the simplex algorithm, our tableau looks like:

Current Tableau:

	RHS	x_1	x_2	x_3	x_4	x_5	x_6	x_7
	-8	0	0	6	0	-1	-3	7
$(x_2 =)$	5	0	1	-2	0	-5	1	-2
$(x_4 =)$	1	0	0	4	1	4	2	2
$(x_1 =)$	2	1	0	-3	0	-3	-2	-1

This was accomplished by adding linear combinations of rows of $Ax = b$ to one another and to the objective function row.

3.3.3 Linear Algebra of Tableaus

SLIDE 10
SLIDE 11

Initial Tableau:

RHS	x
0	c^T
b	A

Current Tableau:

RHS	x
$-\bar{c}_0$	$\bar{c}^T$
$\bar{b}$	$\bar{A}$

Initial Tableau:

RHS	x
0	c^T
b	A

Current Tableau:

RHS	x
$-\bar{c}_0 = 0 - p^T b$	$\bar{c}^T = c^T - p^T A$
$\bar{b} = B^{-1} b$	$\bar{A} = B^{-1} A$

3.3.4 Canonical Form

SLIDE 12

Current Tableau:

	RHS	x_1	x_2	x_3	x_4	x_5	x_6	x_7
	-8	0	0	6	0	-1	-3	7
$(x_2 =)$	5	0	1	-2	0	-5	1	-2
$(x_4 =)$	1	0	0	4	1	4	2	2
$(x_1 =)$	2	1	0	-3	0	-3	-2	-1

This tableau is in canonical form:

- All RHS values are nonnegative.

- For each equation i, there is a variable whose coefficient is $+1$ in this equation and whose coefficient is 0 in all other equations and in the objective function.
- These variables are the basic variables. (x_2, x_4, x_1) (in order) are the basic variables. The other variables are the nonbasic variables (x_3, x_5, x_6, x_7).

3.3.5 Imbedded Identity Matrix

SLIDE 13

Current Tableau:

	RHS	x_1	x_2	x_3	x_4	x_5	x_6	x_7
	-8	0	0	6	0	-1	-3	7
$(x_2 =)$	5	0	1	-2	0	-5	1	-2
$(x_4 =)$	1	0	0	4	1	4	2	2
$(x_1 =)$	2	1	0	-3	0	-3	-2	-1

The ordered equation columns of the basic variables form an identity matrix:

$$\begin{bmatrix} \overline{A}_2 & \overline{A}_4 & \overline{A}_1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

3.3.6 Basic Feasible Solution (b.f.s.)

SLIDE 14

Current Tableau:

	RHS	x_1	x_2	x_3	x_4	x_5	x_6	x_7
	-8	0	0	6	0	-1	-3	7
$(x_2 =)$	5	0	1	-2	0	-5	1	-2
$(x_4 =)$	1	0	0	4	1	4	2	2
$(x_1 =)$	2	1	0	-3	0	-3	-2	-1

The *basic feasible solution* (b.f.s.) corresponding to this tableau is to set all non-basic variables to 0, and all basic variables to their RHS values.

$$\begin{pmatrix} x_2 \\ x_4 \\ x_1 \end{pmatrix} = \begin{pmatrix} 5 \\ 1 \\ 2 \end{pmatrix} \qquad x_3 = x_5 = x_6 = x_7 = 0 \ .$$

This solution satisfies the equations $Ax = b$.
This solution satisfies $x \geq 0$.
The objective function value of the b.f.s. is $z := -(-8) + 6x_3 - 1x_5 - 3x_6 + 7x_7 = 8$.

3.3.7 Optimality Criterion

SLIDE 15

Current Tableau:

	RHS	x_1	x_2	x_3	x_4	x_5	x_6	x_7
	-8	0	0	6	0	-1	-3	7
$(x_2 =)$	5	0	1	-2	0	-5	1	-2
$(x_4 =)$	1	0	0	4	1	4	2	2
$(x_1 =)$	2	1	0	-3	0	-3	-2	-1

If all objective coefficients $\bar{c}_j$ of the nonbasic variables are nonnegative, the current b.f.s. is optimal.
Why?

3.3.8 Min-ratio Test

SLIDE 16

Current Tableau:

	RHS	x_1	x_2	x_3	x_4	x_5	x_6	x_7
	-8	0	0	6	0	-1	-3	7
$(x_2 =)$	5	0	1	-2	0	-5	1	-2
$(x_4 =)$	1	0	0	4	1	4	2	2
$(x_1 =)$	2	1	0	-3	0	-3	-2	-1

- Find a nonbasic variable x_j (column j) whose $\bar{c}_j < 0$. ($j = 6$ here)
- Increase x_j and adjust all basic variables accordingly, until some basic variable becomes 0.

$$\begin{pmatrix} x_2 \\ x_4 \\ x_1 \end{pmatrix} = \begin{pmatrix} 5 \\ 1 \\ 2 \end{pmatrix} - \begin{pmatrix} 1 \\ 2 \\ -2 \end{pmatrix} x_6 = \bar{b} - \overline{A}_6 x_6$$

- This will happen when $x_6 = \theta^* = \min_{\overline{A}_{i6}>0} \left\{ \frac{\bar{b}_i}{\overline{A}_{i6}} \right\} = \min \left\{ \frac{5}{1}, \frac{1}{2}, \frac{2}{-2}\!\!\!\!\!\times \right\} = \frac{1}{2}$

3.3.9 Pivot Operation

SLIDE 17

Current Tableau:

	RHS	x_1	x_2	x_3	x_4	x_5	x_6	x_7
	-8	0	0	6	0	-1	-3	7
$(x_2 =)$	5	0	1	-2	0	-5	1	-2
$(x_4 =)$	1	0	0	4	1	4	2	2
$(x_1 =)$	2	1	0	-3	0	-3	-2	-1

- We reflect the fact that x_6 is now positive (basic) and x_4 is now 0 (nonbasic) by doing row operations to make x_6 a basic variable and x_4 a nonbasic variable.
- We do this by making x_6 the basic variable in the row where x_4 was basic, which is row 2.
- We pivot on $\overline{A}_{26} = 2$.

SLIDE 18

Current Tableau:

	RHS	x_1	x_2	x_3	x_4	x_5	x_6	x_7
	-8	0	0	6	0	-1	-3	7
$(x_2 =)$	5	0	1	-2	0	-5	1	-2
$(x_4 =)$	1	0	0	4	1	4	2	2
$(x_1 =)$	2	1	0	-3	0	-3	-2	-1

New Tableau:

	RHS	x_1	x_2	x_3	x_4	x_5	x_6	x_7
	$-13/2$	0	0	12	$3/2$	5	0	10
$(x_2 =)$	$9/2$	0	1	-4	$-1/2$	-7	0	-3
$(x_6 =)$	$1/2$	0	0	2	$1/2$	2	1	1
$(x_1 =)$	3	1	0	1	1	1	0	1

Is the b.f.s. here optimal? Why?

3.4 The Simplex Algorithm

3.4.1 Termination

SLIDE 19

- The simplex algorithm will only terminate with an optimal b.f.s. or with a demonstration of unboundedness of the objective function.
- Assume that no RHS values ever become zero in the algorithm; then the algorithm improves the objective function at each iteration.
- There are only a finite number of possible b.f.s.'s.
- The simplex algorithm must terminate in a finite number of steps.

3.4.2 Linear Algebra of Tableaus

SLIDE 20

SLIDE 21

Initial Tableau:

RHS	x
0	c^T
b	A

Current Tableau:

RHS	x
$-\bar{c}_0$	$\bar{c}^T$
$\bar{b}$	$\bar{A}$

Initial Tableau:

RHS	x
0	c^T
b	A

Current Tableau:

RHS	x
$-\bar{c}_0 = 0 - p^T b$	$\bar{c}^T = c^T - p^T A$
$\bar{b} = B^{-1} b$	$\bar{A} = B^{-1} A$

for some vector p (called the "simplex multipliers")
and some matrix B (the basis matrix)

3.4.3 The basis matrix B

SLIDE 22

What is B?

Let the basic variable for equation i be denoted $B(i)$.

Then
$$B = \left[\begin{array}{c|c|c|c} A_{B(1)} & A_{B(2)} & \ldots & A_{B(m)} \end{array} \right].$$

SLIDE 23

Initial Tableau:

RHS	x_1	x_2	x_3	x_4	x_5	x_6	x_7
0	-1	1	27	5	17	10	16
4	-1	1	5	1	2	5	1
3	1	0	1	1	1	0	1
9	1	1	3	2	0	3	1

New Tableau:

	RHS	x_1	x_2	x_3	x_4	x_5	x_6	x_7
	$-13/2$	0	0	12	$3/2$	5	0	10
$(x_2 =)$	$9/2$	0	1	-4	$-1/2$	-7	0	-3
$(x_6 =)$	$1/2$	0	0	2	$1/2$	2	1	1
$(x_1 =)$	3	1	0	1	1	1	0	1

$B(1) = 2$, $B(2) = 6$, $B(3) = 1$ Slide 24

Initial Tableau:

RHS	x_1	x_2	x_3	x_4	x_5	x_6	x_7
0	-1	1	27	5	17	10	16
4	-1	1	5	1	2	5	1
3	1	0	1	1	1	0	1
9	1	1	3	2	0	3	1

$B(1) = 2$, $B(2) = 6$, $B(3) = 1$

$$B = \begin{bmatrix} A_2 & A_6 & A_1 \end{bmatrix} = \begin{bmatrix} 1 & 5 & -1 \\ 0 & 0 & 1 \\ 1 & 3 & 1 \end{bmatrix} \qquad B^{-1} = \begin{bmatrix} -\frac{3}{2} & -4 & \frac{5}{2} \\ \frac{1}{2} & 1 & -\frac{1}{2} \\ 0 & 1 & 0 \end{bmatrix}$$

3.5 Simplex Multipliers p

3.5.1 Matrix Algebra of Tableaus

Slide 25

Slide 26

Initial Tableau:

RHS	x
0	c^T
b	A

New Tableau:

RHS	x
$-\bar{c}_0$	$\bar{c}^T$
$\bar{b}$	$\bar{A}$

Initial Tableau:

RHS	x
0	c^T
b	A

New Tableau:

RHS	x
$-\bar{c}_0 = 0 - p^T b$	$\bar{c}^T = c^T - p^T A$
$\bar{b} = B^{-1} b$	$\bar{A} = B^{-1} A$

for some vector p (called the "simplex multipliers")
and some matrix B (the basis matrix)

3.5.2 The Basis

SLIDE 27

Initial Tableau:

RHS	x_1	x_2	x_3	x_4	x_5	x_6	x_7
0	−1	1	27	5	17	10	16
4	−1	1	5	1	2	5	1
3	1	0	1	1	1	0	1
9	1	1	3	2	0	3	1

New Tableau:

	RHS	x_1	x_2	x_3	x_4	x_5	x_6	x_7
	−13/2	0	0	12	3/2	5	0	10
$(x_2 =)$	9/2	0	1	−4	−1/2	−7	0	−3
$(x_6 =)$	1/2	0	0	2	1/2	2	1	1
$(x_1 =)$	3	1	0	1	1	1	0	1

$B(1) = 2$, $B(2) = 6$, $B(3) = 1$

3.5.3 Definition of c_B

SLIDE 28

Initial Tableau:

RHS	x_1	x_2	x_3	x_4	x_5	x_6	x_7
0	−1	1	27	5	17	10	16
4	−1	1	5	1	2	5	1
3	1	0	1	1	1	0	1
9	1	1	3	2	0	3	1

$B(1) = 2$, $B(2) = 6$, $B(3) = 1$

$$c_B^T := \left[c_{B(1)}, \ldots, c_{B(m)}\right] = [c_2, c_6, c_1] = [1, 10, -1]$$

3.5.4 Definition of p

SLIDE 29

p is the solution of:

$$p^T B = c_B^T$$

This is just:

$$p^T = c_B^T B^{-1}$$

In our example, then,

$$p^T = c_B^T B^{-1} = \begin{bmatrix} 1 & 10 & -1 \end{bmatrix} \begin{bmatrix} -\frac{3}{2} & -4 & \frac{5}{2} \\ \frac{1}{2} & 1 & -\frac{1}{2} \\ 0 & 1 & 0 \end{bmatrix} = \begin{bmatrix} \frac{7}{2} & 5 & -\frac{5}{2} \end{bmatrix}$$

p is called the vector of *simplex multipliers*.

3.5.5 Matrix Computations

SLIDE 30

- The mathematics of the simplex algorithm relies on manipulations of A, b, c involving

$$\overline{A} \leftarrow B^{-1}A$$
$$\bar{b} \leftarrow B^{-1}b$$
$$\bar{c} \leftarrow c - p^T A \qquad \text{where } p^T = c_B^T B^{-1}$$

  and where B is always a particular submatrix of A.

3.5.6 Sparsity of A

SLIDE 31

- B is a particular submatrix of A and will be sparse if A is sparse.
- B^{-1} will generally be sparse if B is sparse.
- $B^{-1}A$ will generally be sparse if A and B are sparse.
- Therefore, the key to efficient computation of the simplex method will be the sparsity of A, which will impact on B, B^{-1}, p, and $B^{-1}A$.

4 Variables with Lower and Upper Bounds

SLIDE 32

Most linear optimization models have the following form:

$$\begin{array}{lrcl} \text{minimize} & z = c^T x & & \\ \text{s.t.} & Ax & = & b \\ & l \leq x \leq u & & \end{array}$$

We could convert this to standard form by defining $y := x - l$ and $w := u - x$, obtaining:

$$\begin{array}{lrcrcl} \text{minimize} & z = c^T l & +c^T y & + & 0^T w & & \\ \text{s.t.} & & Ay & + & 0w & = & b - Al \\ & & Iy & + & Iw & = & u - l \\ & & y \geq 0 & & w \geq 0 & & \end{array}$$

SLIDE 33

$$\begin{array}{lrcl} \text{minimize} & z = c^T x & & \\ \text{s.t.} & Ax & = & b \\ & l \leq x \leq u & & \end{array}$$

$$\begin{array}{lrcrcl} \text{minimize} & z = c^T l & +c^T y & + & 0^T w & & \\ \text{s.t.} & & Ay & + & 0w & = & b - Al \\ & & Iy & + & Iw & = & u - l \\ & & y \geq 0 & & w \geq 0 & & \end{array}$$

The LP matrix has gone from $m \times n$ to $(n + m) \times (2n)$

4.1 Example

SLIDE 34

$$\begin{array}{lrcl} \text{minimize} & z = c^T x & & \\ \text{s.t.} & Ax & = & b \\ & l \leq x \leq u & & \end{array}$$

Example:

$$c^T = \begin{bmatrix} 0 & 0 & -2 & -1 & 1 \end{bmatrix}$$

$$A = \begin{bmatrix} 1 & 0 & 1 & -2 & 0 \\ 0 & 1 & -1 & 1 & 2 \end{bmatrix}, b = \begin{bmatrix} 4 \\ 9 \end{bmatrix}$$

$$l = \begin{bmatrix} 1 & 2 & 2 & 1 & 2 \end{bmatrix}$$

$$u = \begin{bmatrix} 5 & 8 & 3 & 3 & 5 \end{bmatrix}$$

4.1.1 Tableau Form

SLIDE 35

$$\begin{array}{lll} \text{minimize } z = & c^T x & \\ \text{s.t.} & Ax & = b \\ & l \le x \le u & \end{array}$$

Form Tableau:

UB	5	8	3	3	5
LB	1	2	2	1	2
RHS	x_1	x_2	x_3	x_4	x_5
0	0	0	-2	-1	1
4	1	0	1	-2	0
9	0	1	-1	1	2

4.1.2 Basic Feasible Solution

SLIDE 36

UB	5	8	3	3	5
LB	1	2	2*	1*	2*
RHS	x_1	x_2	x_3	x_4	x_5
0	0	0	-2	-1	1
4	1	0	1	-2	0
9	0	1	-1	1	2

x_1, x_2 are basic. x_3, x_4, x_5 are at their lower bounds.

$$\begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} 4 \\ 9 \end{pmatrix} - \begin{pmatrix} 1 \\ -1 \end{pmatrix}(2) - \begin{pmatrix} -2 \\ 1 \end{pmatrix}(1) - \begin{pmatrix} 0 \\ 2 \end{pmatrix}(2) = \begin{pmatrix} 4 \\ 6 \end{pmatrix}$$

Is this solution feasible? Yes. Why?
Is this solution optimal? No. Why?

4.1.3 Improving a Solution

SLIDE 37

Let us increase x_3 to $x_3 = 2 + \theta$

$$\begin{array}{lll} x_1 & = & 4 - \theta \\ x_2 & = & 6 + \theta \end{array}$$

We need
$$\begin{array}{ccccccc} 1 & \le & x_1 & = & 4 - \theta & \le & 5 \\ 2 & \le & x_2 & = & 6 + \theta & \le & 8 \\ 2 & \le & x_3 & = & 2 + \theta & \le & 3 \end{array}$$

Largest value of θ is $\theta = 1$, at which point x_3 attains its upper bound.

4.1.4 The New Tableau

SLIDE 38

New tableau is :

UB	5	8	3*	3	5
LB	1	2	2	1*	2*
RHS	x_1	x_2	x_3	x_4	x_5
0	0	0	-2	-1	1
4	1	0	1	-2	0
9	0	1	-1	1	2

4.1.5 Another Iteration

SLIDE 39

UB	5	8	3*	3	5
LB	1	2	2	1*	2*
RHS	x_1	x_2	x_3	x_4	x_5
0	0	0	−2	−1	1
4	1	0	1	−2	0
9	0	1	−1	1	2

x_1, x_2 are basic.
x_3, x_4, x_5 are at one of their bounds, as indicated.

$$\begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} 4 \\ 9 \end{pmatrix} - \begin{pmatrix} 1 \\ -1 \end{pmatrix}(3) - \begin{pmatrix} -2 \\ 1 \end{pmatrix}(1) - \begin{pmatrix} 0 \\ 2 \end{pmatrix}(2) = \begin{pmatrix} 3 \\ 7 \end{pmatrix}$$

Is this solution feasible? Yes. Why?
Is this solution optimal? No. Why?

SLIDE 40

Let us increase x_4 to $x_4 = 1 + \theta$

$$\begin{array}{rcl} x_1 & = & 3 + 2\theta \\ x_2 & = & 7 - \theta \end{array}$$

We need
$$\begin{array}{ccccccc} 1 & \leq & x_1 & = & 3 + 2\theta & \leq & 5 \\ 2 & \leq & x_2 & = & 7 - \theta & \leq & 8 \\ 1 & \leq & x_4 & = & 1 + \theta & \leq & 3 \end{array}$$

Largest value of θ is $\theta = 1$, at which point x_1 reaches its upper bound.

We pivot to remove x_1 from the basis, replacing x_1 by x_4:

SLIDE 41

UB	5	8	3*	3	5
LB	1	2	2	1*	2*
RHS	x_1	x_2	x_3	x_4	x_5
0	0	0	−2	−1	1
4	1	0	1	**−2**	0
9	0	1	−1	1	2

UB	5*	8	3*	3	5
LB	1	2	2	1	2*
RHS	x_1	x_2	x_3	x_4	x_5
−2	−1/2	0	−5/2	0	1
−2	−1/2	0	−1/2	1	0
11	1/2	1	−1/2	0	2

4.1.6 Optimality Criterion

SLIDE 42

UB	5*	8	3*	3	5
LB	1	2	2	1	2*
RHS	x_1	x_2	x_3	x_4	x_5
−2	−1/2	0	−5/2	0	1
−2	−1/2	0	−1/2	1	0
11	1/2	1	−1/2	0	2

x_4, x_2 are basic.
x_1, x_3, x_5 are nonbasic at the bounds indicated above.

$$\begin{pmatrix} x_4 \\ x_2 \end{pmatrix} = \begin{pmatrix} -2 \\ 11 \end{pmatrix} - \begin{pmatrix} -1/2 \\ 1/2 \end{pmatrix} 5 - \begin{pmatrix} -1/2 \\ -1/2 \end{pmatrix} 3 - \begin{pmatrix} 0 \\ 2 \end{pmatrix} 2 = \begin{pmatrix} 2 \\ 6 \end{pmatrix}$$

Is this solution feasible? Yes. Why?
Is this solution optimal? Yes. Why?

4.2 Remarks

SLIDE 43

- The simplex method can be adopted to handle variables with upper and lower bounds, with *no increase* in the number of *rows* or *columns* of the tableau.
- This is very important for computation.

4.3 A Radiation Therapy Model

5 Linear Optimization

5.1 Interior Point Methods

SLIDE 44

- Linear optimization models are solved by either the simplex algorithm or by an interior-point method (IPM).
- Depending on certain aspects of the model, an IPM might be the method of choice.
- We will learn more about IPMs in the second half of the course.

6 Efficiently Updating B^{-1}

6.1 Equations Involving B

6.1.1 The Basis Matrix B

SLIDE 45

At each iteration of the simplex method, we have a basis:

$$B(1), \ldots, B(m) \ ,$$

We form the basis matrix B:

$$B := \left[\ A_{B(1)} \ \middle| \ A_{B(2)} \middle| \ \ldots \middle| \ A_{B(m-1)} \ \middle| \ A_{B(m)} \ \right] \ .$$

6.1.2 Equations Systems to Solve

SLIDE 46

We need to be able to compute:

$$x = B^{-1} r_1 \qquad \text{and/or} \qquad p^T = r_2^T B^{-1} \ ,$$

for iteration-specific vectors r_1 and r_2. Equivalently, solve for x and p:

$$Bx = r_1 \qquad \text{and/or} \qquad p^T B = r_2^T$$

6.2 LU Factorization

6.2.1 Solving $Bx = r_1$

SLIDE 47

Factorize B:

$$B = LU$$

where L, U are lower and upper triangular.

To solve $Bx = r_1$, we compute as follows:

- First solve $Lv = r_1$ for v
- Next solve $Ux = v$ for x.

Then $Bx = LUx = Lv = r_1$

6.2.2 Solving $p^T B = r_2^T$

SLIDE 48

$$B = LU$$

To solve $p^T B = r_2^T$, we compute as follows:

- First solve $u^T U = r_2^T$ for u
- Next solve $p^T L = u^T$ for p.

Then $p^T B = p^T LU = u^T U = r_2^T$

6.3 Rank-1 Matrices

SLIDE 49

$$W = \begin{pmatrix} -2 & 2 & 0 & -3 \\ -4 & 4 & 0 & -6 \\ -14 & 14 & 0 & -21 \\ 10 & -10 & 0 & 15 \end{pmatrix} = \begin{pmatrix} 1 \\ 2 \\ 7 \\ -5 \end{pmatrix} \times \begin{pmatrix} -2 & 2 & 0 & -3 \end{pmatrix} \ .$$

W is an example of rank-1 matrix.
Define

$$u = \begin{pmatrix} 1 \\ 2 \\ 7 \\ -5 \end{pmatrix} \quad \text{and} \quad v^T = \begin{pmatrix} -2 & 2 & 0 & -3 \end{pmatrix} \ .$$

Think of u and v as $n \times 1$ matrices

$$W = uv^T \ .$$

Any rank-1 matrix can be written as uv^T for suitable vectors u and v.

6.4 Rank-1 Update Matrix

6.4.1 Sherman-Morrison Formula

SLIDE 50

Let M be a matrix.
Suppose that we "know" M^{-1}: we have stored M^{-1} or we have an efficient subroutine that solves $Mx = b$ for any RHS b.
Let $\tilde{M} = M + uv^T$.
The Sherman-Morrison Formula: $\tilde{M}$ is invertible if and only if $v^T M^{-1} u \neq -1$, in which case

$$\tilde{M}^{-1} = \left[I - \frac{M^{-1} uv^T}{1 + v^T M^{-1} u} \right] M^{-1} \ .$$

6.4.2 Proof of Formula

SLIDE 51

$$\begin{aligned}
\tilde{M} \times \left[I - \tfrac{M^{-1}uv^T}{1+v^T M^{-1}u} \right] M^{-1} &= \left[M + uv^T \right] \times \left[I - \tfrac{M^{-1}uv^T}{1+v^T M^{-1}u} \right] M^{-1} \\
&= \left[M + uv^T \right] \times \left[M^{-1} - \tfrac{M^{-1}uv^T M^{-1}}{1+v^T M^{-1}u} \right] \\
&= I + uv^T M^{-1} - \tfrac{uv^T M^{-1}}{1+v^T M^{-1}u} - \tfrac{uv^T M^{-1} uv^T M^{-1}}{1+v^T M^{-1}u} \\
&= I + uv^T M^{-1} \left(1 - \tfrac{1}{1+v^T M^{-1}u} - \tfrac{v^T M^{-1}u}{1+v^T M^{-1}u} \right) \\
&= I
\end{aligned}$$

q.e.d.

6.5 Solving Equations with $\tilde{M}^{-1}$

SLIDE 52

We "know" M^{-1}: we have an efficient subroutine that solves $Mx = b$ for any RHS b.
We wish to instead solve $\tilde{M}x = b$
where $\tilde{M} = M + uv^T$

SLIDE 53

$$x = \tilde{M}^{-1} b = \left[I - \frac{M^{-1} uv^T}{1 + v^T M^{-1} u} \right] M^{-1} b \ .$$

Define:

$$x^1 = M^{-1} b \ \text{ and } \ x^2 = M^{-1} u \ ,$$

Then:

$$x = \left[I - \frac{M^{-1} uv^T}{1 + v^T M^{-1} u} \right] x^1 = x^1 - x^2 \left(\frac{v^T x^1}{1 + v^T x^2} \right)$$

SLIDE 54

Procedure for solving $\tilde{M}x = b$:

- Solve the system $Mx^1 = b$ for x^1
- Solve the system $Mx^2 = u$ for x^2
- Compute $x = x^1 - \frac{v^T x^1}{1+v^T x^2} x^2$

6.5.1 Computational Efficiency

SLIDE 55

- n^3 operations to form an LU factorization of M
- n^2 operations to solve $Mx = b$ using back substitution
- $n^3 + n^2$ operations to solve $\tilde{M}x = b$ by factorizing $\tilde{M}$ and then doing back substitution
- $2n^2 + 3n$ if we use the rank-1 update method: we need to do 2 back substitution solves, and then $3n$ operations for the final step
- This is vastly superior to $n^3 + n^2$ for large n

6.6 Updating B and B^{-1}

6.6.1 Updating the Basis

SLIDE 56

Assume that the columns of A have been re-ordered so that

$$B := [\ A_1 \ | \ldots | \ A_{j-1} \ | \ A_j \ | \ A_{j+1} \ | \ \ldots \ | \ A_m \ ]$$

at one iteration. At the next iteration we have a new basis matrix $\tilde{B}$:

$$\tilde{B} := [\ A_1 \ | \ldots | \ A_{j-1} \ | \ A_k \ | \ A_{j+1} \ | \ \ldots \ | \ A_m \ ] \ .$$

Column A_j has been replaced by column A_k in the new basis.

6.6.2 Basis Update Formula

SLIDE 57

$$B := [\ A_1 \ | \ldots | \ A_{j-1} \ | \ A_j \ | \ A_{j+1} \ | \ \ldots \ | \ A_m \ ]$$
$$\tilde{B} := [\ A_1 \ | \ldots | \ A_{j-1} \ | \ A_k \ | \ A_{j+1} \ | \ \ldots \ | \ A_m \ ]$$

Note that $\tilde{B} = B + (A_k - A_j) \times \left(e^j\right)^T$
e^j is the j^{th} unit vector
$\tilde{B} = B + uv^T$ with

$$u = (A_k - A_j) \quad \text{and} \quad v = \left(e^j\right)$$

SLIDE 58

To solve the equation system $\tilde{B}x = r_1$, we can apply the rank-1 update method, substituting $M = B$, $b = r_1$, $u = (A_k - A_j)$ and $v = \left(e^j\right)$. This works out to:

- Solve the system $Bx^1 = r_1$ for x^1
- Solve the system $Bx^2 = A_k - A_j$ for x^2
- Compute $x = x^1 - \frac{(e^j)^T x^1}{1 + (e^j)^T x^2} x^2$

What if we want to do this over a sequence of iterations?

6.7 More Algebraic Manipulation

SLIDE 59

$$\tilde{B}^{-1} = \left[I - \frac{B^{-1}uv^T}{1+v^T B^{-1}u}\right] B^{-1}$$

$$= \left[I - \frac{B^{-1}(A_k - A_j)(e^j)^T}{1+(e^j)^T B^{-1}(A_k - A_j)}\right] B^{-1}$$

But $A_j = Be^j$, whereby $B^{-1}A_j = e^j$

$$\tilde{B}^{-1} = \left[I - \frac{(B^{-1}A_k - e^j)(e^j)^T}{(e^j)^T B^{-1}A_k}\right] B^{-1} = \tilde{E}B^{-1}$$

where

$$\tilde{E} = \left[I - \frac{\left(B^{-1}A_k - e^j\right)\left(e^j\right)^T}{(e^j)^T B^{-1}A_k}\right]$$

SLIDE 60

$\tilde{B}^{-1} = \tilde{E}B^{-1}$ where $\tilde{E} = \left[I - \frac{(B^{-1}A_k - e^j)(e^j)^T}{(e^j)^T B^{-1}A_k}\right]$

Procedure for solving the system $\tilde{B}x = r_1$:

- Solve the system $B\tilde{w} = A_k$ for $\tilde{w}$
- Form and save the matrix $\tilde{E} = \left[I - \frac{(\tilde{w} - e^j)(e^j)^T}{(e^j)^T \tilde{w}}\right]$
- Solve the system $Bx^1 = r_1$ for x^1
- Compute $x = \tilde{E}x^1$

SLIDE 61

$$\tilde{E} = \begin{pmatrix} 1 & & & \tilde{c}_1 & & \\ & 1 & & \tilde{c}_2 & & \\ & & \ddots & \vdots & & \\ & & & \tilde{c}_j & & \\ & & & \vdots & \ddots & \\ & & & \tilde{c}_m & & 1 \end{pmatrix} \quad \text{with } \tilde{c} = \frac{\left(\tilde{w} - e^j\right)}{(e^j)^T \tilde{w}}$$

$\tilde{E}$ is an elementary matrix

We store $\tilde{E}$ by only storing the column $\tilde{c}$ and the index j

6.8 Implementation over Sequential Iterations

SLIDE 62

$$\tilde{B} := [\ A_1\ \mid \ldots \mid A_{i-1}\ \mid A_i \mid\ A_{i+1} \mid\ \ldots \mid\ A_m\ ]$$

At the next iteration, we replace the column A_i with the column A_l :

$$\tilde{\tilde{B}} := [\ A_1\ \mid \ldots \mid A_{i-1}\ \mid A_l \mid\ A_{i+1} \mid\ \ldots \mid\ A_m\ ]\ .$$

SLIDE 63

Let $\tilde{\tilde{w}}$ be the solution of the system $\tilde{B}\tilde{\tilde{w}} = A_i$

$$\tilde{\tilde{B}}^{-1} = \tilde{\tilde{E}}\tilde{B}^{-1}$$

where

$$\tilde{\tilde{E}} = \left[I - \frac{\left(\tilde{\tilde{w}} - e^i\right)\left(e^i\right)^T}{(e^i)^T \tilde{\tilde{w}}}\right]$$

- $\tilde{\tilde{B}}^{-1} = \tilde{\tilde{E}}\tilde{B}^{-1} = \tilde{\tilde{E}}\tilde{E}B^{-1}$
- We solve equations involving $\tilde{\tilde{B}}$ by forming $\tilde{E}$, $\tilde{\tilde{E}}$ and the LU factorization of B.

SLIDE 64

- We start with a basis B and we compute and store an LU factorization of B.
- Our sequence of bases is $B_0 = B, B_1, \ldots, B_k$
- We compute matrices $E_1, \ldots, E_k$ with the property that

$$(B_l)^{-1} = E_l E_{l-1} \cdots E_1 B^{-1} \quad , \ l = 1, \ldots, k \ .$$

- For the next basis inverse B_{k+1} we compute a new matrix E_{k+1} and we write:

$$(B_{k+1})^{-1} = E_{k+1} E_k \cdots E_1 B^{-1}$$

6.9 Recursive Implementation

SLIDE 65

- The details for implementing this scheme are straightforward
- The notation is not much fun
- After $K = 50$ or so pivots of applying the above methodology, round-off error tends to accumulate
- Most simplex codes do a complete basis re-factorization every $K = 50$ pivots.