

Math 340, Nov 16

Idea: don't have to write the whole dictionary...

don't write all this

$$\begin{cases} X_B \\ \vdots \\ \vdots \\ \vdots \end{cases} = \begin{pmatrix} A_B^{-1} \vec{b} \\ \vdots \\ \vdots \\ \vdots \end{pmatrix} = A_B^{-1} A_N \vec{x}_N$$

$$z = - \left(\vec{c}_N - \vec{c}_B A_B^{-1} A_N \right) \vec{x}_N$$

=
Start $\max \vec{c} \cdot \vec{x}$ s.t.
 $A \vec{x} \leq \vec{b}$
 $\vec{x} \geq 0$

\vec{x} n-dimensional, $A = A_{orig}$ is an $m \times n$ matrix, \vec{b} m-dimensional

What is the revised simplex method?

① Do this, do that - - it does well... 😊

Truth: Revised simplex method only works well in certain cases...

$$X_B = \text{blah blah blah}$$

$$z = \text{blah blah} \left(\text{blah} - \underbrace{\vec{c}_B^T A_B^{-1} A_N}_{\text{key to the whole method...}} \right) \vec{x}_N$$

Rough idea:

① Focus on $z = - \left(\vec{c}_B^T A_B^{-1} A_N \right) \vec{x}_N$

$$\left(\dots \quad \vec{c}_B^T \cdot A_B^{-1} \cdot A_N \right) \vec{x}_N$$

$1 \times m$ row vector $m \times m$ $m \times n$
 serious problem serious problem

② If A_B^{-1} is ~~messy~~ to compute ... 😞

- m small, say $m = n^{1/4}$
 Naive invert $m^3 = n^{3/4}$ 😊

$$\left[A \mid I \right] \begin{pmatrix} \vec{x}_{dec} \\ \vec{x}_{slack} \end{pmatrix} = \vec{b}$$

$$z = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix} \cdot \begin{pmatrix} \vec{x}_{dec} \\ \vec{x}_{slack} \end{pmatrix}$$

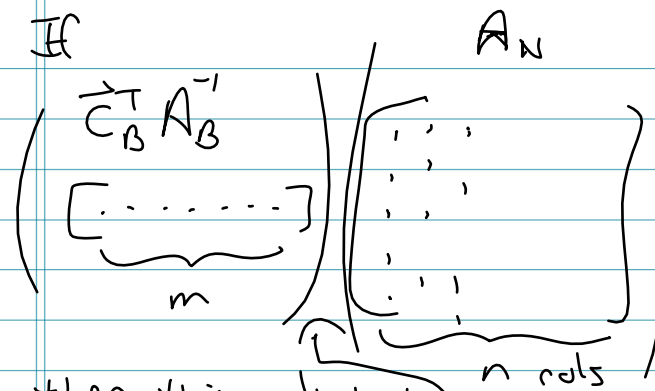
n dec vars, m slack vars

$$\vec{x}_{slack} = \vec{b} - A_{orig} \vec{x}_{dec}$$

basic vars always m

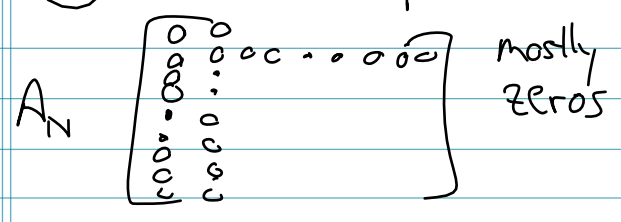
non-basic always n

= Naive simplex method takes about $m \cdot n$ {mults additions} per pivot



then this multiplication takes mn mults mn adds

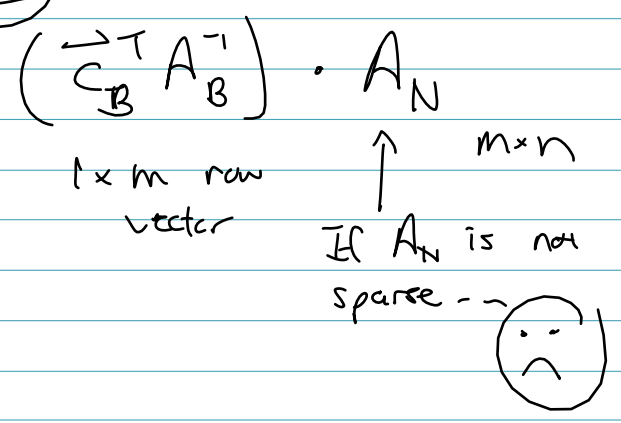
For (II) after A_N is sparse ✓



(I) A_B^{-1} ?

- If A_B has a special structure s.t. A_B^{-1} ~~easy~~ ^{fast} to invert,
- From step to step, A_B and A_B^{-1} doesn't change much

(II) Look at



For (II)

$$z = \dots (C_N^T - C_B^T A_B^{-1} A_N) \vec{x}_N$$

Do we need all of z row?

- Yes, at the final dictionary:

$$z = \dots - 3x_1 - 5x_2 - 20x_3 \dots$$

↑ ↑ ↑
all negatives

But in the middle

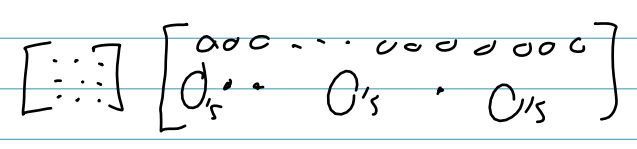
$$z = 20 + 5x_4 + 30x_5 + 20x_6 - 3x_7 - 17x_8 \dots$$

Remark:

$$\vec{x}_B = A_B^{-1} \vec{b} - \underbrace{A_B^{-1} A_N}_{\text{99% of each row \& col is 0's}} \vec{x}_N$$

$m = n^{1/4}$ so

A_B^{-1} no problem...



Have to compare with naive simplex method, mn FLOPs

Rather than evaluating all of

$$C_B^T A_B^{-1} - A_N X_N$$

do it one coefficient at a time

=
What if

$$z = 30 + 2x_1 + 5x_3 - 4x_7 - 9x_6 + 12231x_8 + 21974x_{11}$$

- take x_8, x_{11} into basis
- guess that both into basis does well

$$\left(\begin{array}{c} C_B^T A_B^{-1} \\ \vdots \end{array} \right) A_N \vec{x}_N$$

$\underbrace{\quad}_{1 \times n}$ $\left[\begin{array}{c} \vdots \\ \vdots \\ \vdots \end{array} \right]$

x_5

$$z = \begin{matrix} \text{bleh} \\ \text{bleh} \\ \text{bleh} \end{matrix} \quad \textcircled{+30x_5} \quad \begin{matrix} \text{bleh} \\ \text{bleh} \\ \text{bleh} \end{matrix}$$

x_5 can enter...

$$\textcircled{II} \quad \left(C_B^T A_B^{-1} \right) A_N \vec{x}_N$$

If you know vector
to find a z-row coef for each
takes just m mult add for each
coef

Other possibility:

evaluate all of z row,
this give you guess as to which
to try next few times

then

- one pivot: find all z row
- next few pivots find $C_B^T A_B^{-1} A_N X_N$ at a few like candidates to enter basis

Next time ... A_B^{-1} ...