CS 433 Automated Reasoning 2022

Lecture 16: Theory of linear rational arithmetic (LRA)

Instructor: Ashutosh Gupta

IITB, India

Compile date: 2022-09-29

Topic 16.1

Theory of linear rational arithmetic

Linear rational arithmetic (LRA)

Formulas with structure $\Sigma = (\{+/2, 0, 1, \dots\}, \{< /2\})$ with a set of axioms

Note: We have seen the axioms in the third lecture.

Example 16.1

The following formulas are in the quantifier-free fragment of the theory (QF_LRA), where x, y, and z are the rationals.

- $x \geq 0 \vee y + z = 5$
- $x < 300 \wedge x - z \neq 5$

Exercise 16.1

There is no $\leq$ in the signature. How can we use the symbol?

Proof system for QF_LRA

Due to the Farkas lemma, the following proof rule is complete for the reasoning over QF_LRA.

$$[\textsc{Comb}]\frac{t_1 \leq 0 \quad t_2 \leq 0}{t_1\lambda_1 + t_2\lambda_2 - \lambda_3 \leq 0}\lambda_1, \lambda_2, \lambda_3 \geq 0$$

Example 16.2

The following is an instance of the proof step

$$\frac{2x - y \leq 1 \quad 4y - 2x \leq 6}{x + y \leq 5}\lambda_1 = 1, \lambda_2 = 0.5, \lambda_3 = 1$$

Example 16.3

The following is an another instance of the proof step that derives false.

$$\frac{x + y \leq -2 \quad -x \leq 0 \quad -y \leq 1}{0 \leq -1}$$

Flattened rule instances

Theory solver for rational linear arithmetic

We will discuss the following method to find satisfiability of conjunction of linear inequalities.

- Simplex

We may cover some of the following methods in the next lecture.

- Fourier-Motzkin
- Elliposid method
- Kermakar's method

We present the above methods using non-strict linear inequalities. However, they are extendable to strict inequalities, equalities, disequalities.

Topic 16.2

Simplex

Simplex

Simplex was originally designed for linear optimization problems, e.g., $max\{cx|Ax \leq b\}$..

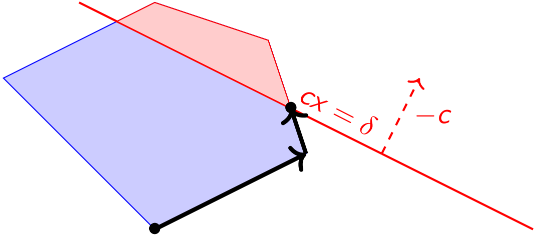
A simplex variation is used to check satisfiablity, called incremental simplex.

Commentary: In fact, there are several design choices for implementing simplex. The presentation here is one version of simplex.

Incremental simplex

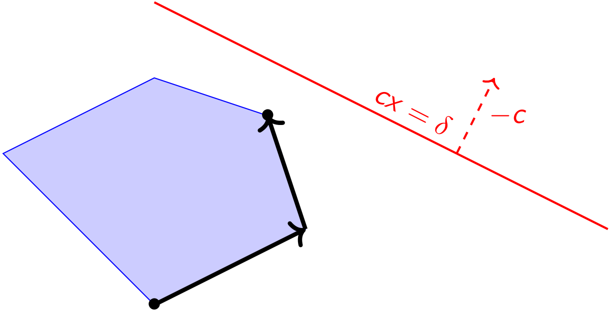
Incremental simplex

- takes atoms one by one,
- maintains a current assignment that satisfies the atoms seen so far, and
- after receiving a new atom $cx \leq \delta$,
  - attempts to move the assignment in the direction of $-c$ (optimization like operation)



Incremental simplex: unsatisfiable input

Simplex may fail to reach $cx = \delta$ and the input is unsatisfiable



Exercise 16.2

Who is responsible for the unsatisfiability?

Incremental simplex as theory solver

Recall the expected interface for SMT solver:

- push(): add new atom to the simplex state.
- pop(): inexpensive operation
- unsatCore(): again inexpensive operation

Topic 16.3

Simplex - terminology

Notation

Consider the conjunction of linear inequalities in matrix form

$$Ax \leq b,$$

where A is a $m \times n$ matrix.

By introducing fresh variables, we transform the above into

$$\begin{bmatrix} -\mathtt{I} & A \end{bmatrix} \begin{bmatrix} s \\ x \end{bmatrix} = 0 \text{ and } s \leq b.$$

s are called slack variables. Since there is no reason to distinguish x and s in simplex, A will refer to $\begin{bmatrix} -\mathtt{I} \; A \end{bmatrix}$ and x will refer to $\left[\begin{smallmatrix} s \\ x \end{smallmatrix}\right]$.

Notation (contd.)

In general, the constraints will be denoted by

$$Ax = 0 \text{ and } \bigwedge_{i=1}^{m+n} l_i \leq x_i \leq u_i.$$

l_i and u_i are $+\infty$ and $-\infty$ if there is no lower and upper bound, respectively.

- A is $m \times (m+n)$ matrix.
- Since $Ax = 0$ defines an n-dim subspace in $(m+n)$-dim space, if we choose values of n variables then we fix values of the other m variables.
- We will refer to ith column of A as the column corresponding to x_i.

Example: notation

Example 16.4

Consider: $-x + y \leq -2 \wedge x \leq 3$

We introduce slack variables s_1 and s_2 for each inequality.

In matrix form,

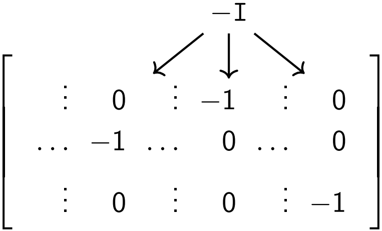
$$\left[\begin{array}{cc:cc} -1 & 0 & -1 & 1 \\ 0 & -1 & 1 & 0 \end{array}\right] \left[\begin{array}{c} s_1 \\ s_2 \\ \hdashline x \\ y \end{array}\right] = 0 \qquad \begin{array}{c} s_1 \leq -2 \\ s_2 \leq 3 \end{array}$$

Basic and nonbasic variables

Definition 16.1

Simplex assumes all the columns of $-\mathtt{I}$ *(of size* $m \times m$*) occur in* A*.*

- *The variables corresponding to the columns are called basic variables.*
- *Others are called nonbasic variables.*



$$-\mathtt{I}$$

$$\begin{bmatrix} \vdots & 0 & \vdots & -1 & \vdots & 0 \\ \dots & -1 & \dots & 0 & \dots & 0 \\ \vdots & 0 & \vdots & 0 & \vdots & -1 \end{bmatrix}$$

Exercise 16.3

What are the numbers of basic and nonbasic variables ?

Example: Basic and nonbasic variables

Definition 16.2

Let B be the set of indexes for the basic variables and $NB \triangleq 1..(m+n) - B$. For $j \in B$, let k_j be a row such that $A_{k_j j} = -1$ and we may write

$$x_j = \sum_{i \in NB} a_{k_j i} x_i,$$

which is called the definition of x_j.

Example 16.5

$$\begin{bmatrix} -1 & 0 & -1 & 1 \\ 0 & -1 & 1 & 0 \end{bmatrix} \begin{bmatrix} s_1 \\ s_2 \\ x \\ y \end{bmatrix} = 0 \qquad \begin{array}{c} s_1 \leq -2 \\ s_2 \leq 3 \end{array}$$

Currently, s_1 and s_2 are basic and x and y are nonbasic. $B = \{1, 2\}$, $NB = \{3, 4\}$, $k_1 = 1$, and $k_2 = 2$. The definition of s_1 is $-x + y$.

Exercise 16.4

What is the definition of the other basic variable?

Current assignment

Definition 16.3

Simplex maintains current assignment $v : x \to \mathbb{Q}$ such that

- $Av = 0$,
- *nonbasic variables satisfy their bounds, and,*
- *consequently values for basic variables in v are fixed and v may violate a bound of at most one basic variable.*

Explained later why "at most" one

Example 16.6

$$\left[\begin{array}{cc:cc} -1 & 0 & -1 & 1 \\ 0 & -1 & 1 & 0 \end{array}\right] \left[\begin{array}{c} s_1 \\ s_2 \\ \hdashline x \\ y \end{array}\right] = 0$$

Currently violated

$$s_1 \leq -2$$
$$s_2 \leq 3$$

Initially, $v = \{\underbrace{x \mapsto 0, y \mapsto 0}_{\text{Choose values for nonbasic variables, others follow!}}, s_1 \mapsto 0, s_2 \mapsto 0\}$

State

Simplex ensures the following invariant.

For variable $i \in NB$,

- if x_i is unbounded then $v(x_i) = 0$ and
- otherwise $v(x_i)$ is equal to one of the existing bounds of x_i

Definition 16.4

A bound on x_i is called active if $v(x_i)$ is equal to the bound.

We will mark the active bounds by *.

Definition 16.5

The NB set and bound activity defines the current state of simplex.

Example 16.7

$$\begin{bmatrix} -1 & 0 & -1 & 1 \\ 0 & -1 & 1 & 0 \end{bmatrix} \begin{bmatrix} s_1 \\ s_2 \\ x \\ y \end{bmatrix} = 0$$

$$s_1 \leq -2$$
$$s_2 \leq 3$$

Since all nonbasic variables have no bounds, no bound is marked active.

Topic 16.4

Simplex - pivot operation

Pivot operation

If v violates a bound of a basic variable, then simplex corrects it by pivoting.

Definition 16.6

Let us suppose x_j is basic, column j has -1 at row k, and x_i is nonbasic. A pivot operation between i and j *exchanges the role between x_i and x_j, i.e., row operations until column i has a single nonzero entry -1 at row k.*

After pivot operation between i and j

$$\begin{array}{rc} & \begin{array}{ccccc} & j & & i & \\ & \downarrow & & \downarrow & \end{array} \\ k \longrightarrow & \left[\begin{array}{ccccc} \vdots & 0 & \vdots & a & \vdots \\ \dots & -1 & \dots & b & \dots \\ \vdots & 0 & \vdots & c & \vdots \end{array}\right] \end{array} \rightsquigarrow \left[\begin{array}{ccccc} \vdots & \frac{a}{b} & \vdots & 0 & \vdots \\ \dots & \frac{1}{b} & \dots & -1 & \dots \\ \vdots & \frac{c}{b} & \vdots & 0 & \vdots \end{array}\right]$$

Variables for pivot operations

Three variables are involved in the pivoting

1. the violated basic variable
2. nonbasic variable for pivot
3. basic variable for pivot

The violated basic variable does not participate in pivoting.

Commentary: The above claim is not entirely accurate. In a special case, the violated basic variable may participate in pivoting. Otherwise, the violated variable remains basic variables after pivot.

Violated basic variable

Wlog, let $1 \in B$, $k_1 = 1$, and $v(x_1)$ violates u_1.

We need to decrease $v(x_1)$.

We call $v(x_1) - u_1$ violation difference.

Exercise 16.5

Write other cases that are ignored due to "wlog"

Choosing nonbasic column for pivot

Since $x_1 = \sum_{i \in NB} a_{1i} x_i$, we need to change $v(x_i)$ of some x_i such that $a_{1i}x_i$ decreases

Definition 16.7

A column $i \in NB$ is suitable if

- *x_i is unbounded,*
- *$v(x_i) = u_i$ and $a_{1i} > 0$, or*
- *$v(x_i) = l_i$ and $a_{1i} < 0$.*

*i is **selected** suitable column if i is the smallest suitable column.*

Example 16.8

$$\begin{bmatrix} -1 & 0 & -1 & 1 \\ 0 & -1 & 1 & 0 \end{bmatrix} \begin{bmatrix} s_1 \\ s_2 \\ x \\ y \end{bmatrix} = 0 \qquad \begin{array}{c} s_1 \leq -2 \\ s_2 \leq 3 \end{array}$$

Column 3 and 4 are suitable.

Choosing basic column for pivot I

So far: v satisfies all bounds except u_1 and $i \in NB$ is the selected suitable variable.

Since x_i appears also in the definitions of the basic variables, change in $v(x_i)$ may lead to the other violations.

Consider the following definition of $j \in B$.

$$x_j = a_{k_j i} x_i + \sum_{i' \in NB - \{i\}} a_{k_j i'} x_{i'},$$

If $a_{k_j i} \neq 0$, changes in x_i will change x_j.

Since we assume single violation at a time, we have $l_j \leq v(x_j) \leq u_j$.

$$\begin{array}{c}
\begin{array}{cccccccc}
\text{Violated} & & j & & & & i & \\
\downarrow & & \downarrow & & & & \downarrow & \\
\end{array} \\
\begin{bmatrix}
-1 & \dots & 0 & \dots & 0 & \dots & a_{1i} & \dots \\
\vdots & \dots & \vdots & \dots & \vdots & \dots & \vdots & \dots \\
0 & \dots & -1 & \dots & 0 & \dots & a_{k_j i} & \dots \\
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots
\end{bmatrix}
\end{array}$$

Choosing basic column for pivot - available slack

Consider again the following definition of $j \in B$.

$$x_j = a_{k_j i} x_i + \sum_{i' \in NB - \{i\}} a_{k_j i'} x_{i'},$$

If $a_{k_j i} > 0$, if we increase x_i it will increase x_j.

The following amount is the maximum x_i can increase without violating x_j upper bound u_j.

$$\frac{u_j - v(x_j)}{a_{k_j i}}$$

Exercise 16.6

What is the expression for maximum allowed change if $a_{k_j i} < 0$?

Choosing basic column for pivot : index that allows minimum change

Wlog, let $a_{1i} < 0$. Therefore, we need to increase $v(x_i)$.

Definition 16.8

We need to find the maximum allowed change.

$$ch := min\{\frac{u_j - v(x_j)}{a_{k_j i}} | a_{k_j i} > 0 \wedge j \in B\} \cup \{\frac{l_j - v(x_j)}{a_{k_j i}} | a_{k_j i} < 0 \wedge j \in B\}$$

We choose the smallest j for which the above min is attained.

Exercise 16.7

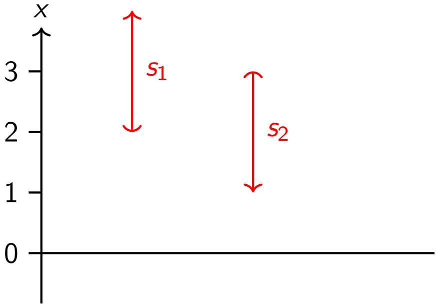
What are the other cases in the without loss of generality?

Example: choosing basic column for pivot

Example 16.9

We change x(selected suitable column) to reduce violation difference.
Since $v(y) = 0$ and we are varying x, $s_1 = -x$ and $s_2 = x$.

The bounds on basic variables are $s_1 \leq -2$, and $s_2 \leq 3$.



Therefore, s_1 allows $2 \leq x$ and s_2 allows $x \leq 3$.

Clearly, $ch = 3$ and $j = 2$.

Simplex - pivoting operation to reduce violation difference

We carry ch and j from the last slide. Wlog, $ch = \frac{u_j - v(x_j)}{a_{k_j i}}$.

Now there are three possibilities

1. If $ch = u_i = +\infty$, pivot between i and 1 and activate u_1
2. If $ch > (u_i - l_i)$, we assign $v(x_i) = l_i$ and no pivoting
3. Otherwise, we apply pivoting between nonbasic i and basic j. We activate u_j bound on variable x_j.

If the violation persists, we apply further pivot operations.

Theorem 16.1

Pivoting operation never increases violation difference

Example: pivoting

Example 16.10

Our running example, s_1 is in violation, chosen nonbasic column is 3 and chosen basic column is 2

$$\begin{bmatrix} -1 & 0 & -1 & 1 \\ 0 & -1 & 1 & 0 \end{bmatrix} \begin{bmatrix} s_1 \\ s_2 \\ x \\ y \end{bmatrix} = 0 \qquad \begin{array}{c} s_1 \leq -2 \\ s_2 \leq 3 \end{array}$$

After pivoting between 3 and 2.

$$\begin{bmatrix} -1 & -1 & 0 & 1 \\ 0 & 1 & -1 & 0 \end{bmatrix} \begin{bmatrix} s_1 \\ s_2 \\ x \\ y \end{bmatrix} = 0 \qquad \begin{array}{c} s_1 \leq -2 \\ s_2 \leq 3^* \end{array}$$

Now v is satisfying.

Exercise 16.8

What is v?

Topic 16.5

Incremental simplex

Incremental simplex and single violation assumption

Before adding next atom, simplex has a solution of atoms added so far.

New atom $cx \leq \delta$ is added in the following steps.

- A fresh slack variable s is introduced
- $s = cx$ is added as a row in A and $s \leq \delta$ is added in the bounds
- The new row may have non-zeros in basic columns. They are removed by row operations on the new row.
- s is added to B, declaring it to be a basic variable.

Therefore, the current assignment can only violate the bound of s.

The above strategy is called eager pivoting. We may lazily remove the violations, without breaking the correctness.

Example: inserting a new atom

Example 16.11

Let us add $-2x - y \leq -8$ in our example. We add a slack variable s_3 and a corresponding row.

$$\begin{bmatrix} -1 & 0 & 0 & -2 & -1 \\ 0 & -1 & -1 & 0 & 1 \\ 0 & 0 & 1 & -1 & 0 \end{bmatrix} \begin{bmatrix} s_3 \\ s_1 \\ s_2 \\ x \\ y \end{bmatrix} = 0 \qquad \begin{array}{r} s_3 \leq -8 \\ s_1 \leq -2 \\ s_2 \leq 3^* \end{array}$$

After removing basic variables $(\{s_1, x\})$ from the top row

$$\begin{bmatrix} -1 & 0 & -2 & 0 & -1 \\ 0 & -1 & -1 & 0 & 1 \\ 0 & 0 & 1 & -1 & 0 \end{bmatrix} \begin{bmatrix} s_3 \\ s_1 \\ s_2 \\ x \\ y \end{bmatrix} = 0 \qquad \begin{array}{r} s_3 \leq -8 \\ s_1 \leq -2 \\ s_2 \leq 3^* \end{array}$$

Exercise 16.9

Now s_3 is violated. Pivot if possible.

Simplex - iterations

Simplex is a sequence of pivot operations

- If a state is reached without violation then v is a satisfying assignment.
- If there are no suitable columns to repair a violation then input is unsat.

Example 16.12

s_3 is still in violation.

$$\begin{bmatrix} -1 & -1 & -3 & 0 & 0 \\ 0 & 1 & 1 & 0 & -1 \\ 0 & 0 & 1 & -1 & 0 \end{bmatrix} \begin{bmatrix} s_3 \\ s_1 \\ s_2 \\ x \\ y \end{bmatrix} = 0 \qquad \begin{array}{c} s_3 \leq -8 \\ s_1 \leq -2^* \\ s_2 \leq 3^* \end{array}$$

Now, we can not find a suitable column.
Therefore, the constraints are unsat.

Example 16.13

Run simplex on $x_1 \leq 5 \wedge 4x_1 + x_2 \leq 25 \wedge -2x_1 - x_2 \leq -25$

After push of the first atom

$$\begin{bmatrix} -1 & 1 \end{bmatrix} \begin{bmatrix} s_1 \\ x_1 \end{bmatrix} = 0 \qquad s_1 \leq 5 \qquad v = \{x_1 \mapsto 0, s_1 \mapsto 0\}$$

After push of the second atom

$$\begin{bmatrix} -1 & 0 & 4 & 1 \\ 0 & -1 & 1 & 0 \end{bmatrix} \begin{bmatrix} s_2 \\ s_1 \\ x_1 \\ x_2 \end{bmatrix} = 0 \qquad \begin{array}{c} s_1 \leq 5 \\ s_2 \leq 25 \end{array} \qquad v = \{\_ \mapsto 0\}$$

After push of the last atom

$$\begin{bmatrix} -1 & 0 & 0 & -2 & -1 \\ 0 & -1 & 0 & 4 & 1 \\ 0 & 0 & -1 & 1 & 0 \end{bmatrix} \begin{bmatrix} s_3 \\ s_2 \\ s_1 \\ x_1 \\ x_2 \end{bmatrix} = 0 \qquad \begin{array}{c} s_1 \leq 5 \\ s_2 \leq 25 \\ s_3 \leq -25 \end{array} \qquad v = \{\_ \mapsto 0\}$$

Exercise 16.10*Finish the run*

Theory solver interface pop()

If we want to remove some atom from simplex state, we

- make the corresponding slack variable x_i basic variable and
- remove the corresponding row k_i and bound constraints on x_i

Cost: one pivot operation

Theory solver interface UnsatCore()

If input is unsat, there must be a violated basic variable x_j

- we collect the slack variables that appear in the row k_j
- the atoms corresponding to the slack variables are part of unsat core

Cost: zero.

However, we used the simplex design that excessively uses slack variables.

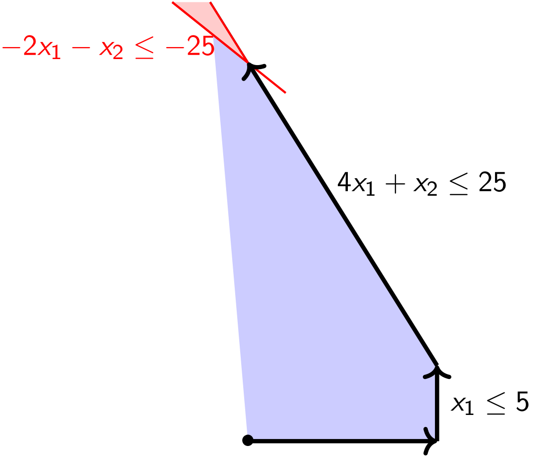
Commentary: Some times slack variables can be avoided. For example, input atom is equality. We can solve the constraints without introducing slack variables.

Topic 16.6

Complexity of simplex

An example of worst case Simplex

The previous example is the case of exponential number of pivots.



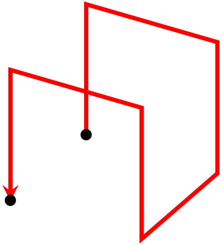
Worst case Simplex

For n dimensional problem, we can make simplex do $2^n - 1$ pivots, i.e., walking on $2^n - 1$ edges of a n-dimensional cuboid.

Example 16.14

Problem input for three dimensions

$x_1 \leq 5 \land 4x_1 + x_2 \leq 25 \land 8x_1 + 4x_2 + x_3 \leq 125 \land -4x_1 - 2x_2 - x_3 \leq -125$



Simplex complexity

Simplex is average time linear and worst case exponential.

In practice, none of the above complexities are observed

Ellipsoid method is a polynomial time algorithm for linear constraints. In practice, simplex performs better in many classes of problems.

Exercise: notation

$$\begin{bmatrix} -1 & 0 & 0 & -2 & -1 \\ 0 & -1 & 0 & 4 & 1 \\ 0 & 0 & -1 & 1 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \end{bmatrix} = 0 \qquad \begin{array}{r} s_1 \leq 5 \\ s_2 \leq 25 \\ s_3 \leq -25 \end{array} \qquad v = \{\_ \mapsto 0\}$$

Topic 16.7

Extra slides : Incremental simplex - geometric intuition

Simplex- geometric intuition

Now we will connect the algorithm with a geometric intuition.

For ease of exposition, we will assume that all l_is and u_is are finite.
This restriction can be easily dropped.

We will add supper script p to various objects to denote their value at pth iteration.

For example, A^p is the value of A at pth iteration.

Simplex - geometric intuition: meaning of suitable column

Let us introduce the following object in each iterations

- Let μ^p be a row vector of length $2(m+n)$ such that

$$\begin{bmatrix} 1 & \underbrace{0}_{m-1} & \mu^p \end{bmatrix} \begin{bmatrix} A^p \\ \mathrm{I} \\ -\mathrm{I} \end{bmatrix} = \begin{bmatrix} -1 & \underbrace{0}_{m+n-1} \end{bmatrix}$$

$$\mu^p_k = \begin{cases} -A^p_{1k} & k \in NB^p \text{ and } u_k \text{ is active at } p\text{th iteration} \\ A^p_{1(k-(m+n))} & (k-(m+n)) \in NB^p \text{ and } l_{k-(m+n)} \text{ is active at } p\text{th iteration} \\ 0 & \text{otherwise} \end{cases}$$

Theorem 16.2

Let i' be the smallest index for which μ^p has a negative number and i be the selected suitable column for the next pivoting. Then,

$$i = \begin{cases} i' & i' \leq m+n \\ i'-(m+n) & \textit{otherwise.} \end{cases}$$

Exercise 16.11

Prove the above.

Simplex - geometric intuition: update direction

Selection of suitable column induces the idea of update direction

- Let y^p be a vector of length $m + n$. y^p indicates the direction of change due to pivot operation after pth iteration.
  Let $i \in NB^p$ be the selected suitable column.
  - l_i is active

$$y^p_j = \begin{cases} 1 & j = i, \\ A^p_{k_j i} & j \in B^p \\ 0 & \text{otherwise} \end{cases}$$

  - u_i is active

$$y^p_j = \begin{cases} -1 & j = i, \\ -A^p_{k_j i} & j \in B^p \\ 0 & \text{otherwise} \end{cases}$$

Exercise 16.12

Show $[-1\ 0]y^p > 0$

Simplex - geometric intuition: limit on update

- The change in direction y only violate bounds on basic variables

$$ch := min \bigcup_{j \in 1..m} \{\frac{u_j - v(x_j)}{y^p_j} | y^p_j > 0\} \cup \{\frac{l_j - v(x_j)}{-y^p_j} | y^p_j > 0\}$$

Let j be the smallest index for which the above min is attained, which is used for pivoting.

Exercise 16.13

Check the basis column j selected above is same as the pivot basis column selected earlier

Simplex - termination

Lemma 16.1

Simplex terminates.

Proof.

In every step the violation difference $(v(x_1) - u_1)$ reduces or stays same.

Since there are finitely many states, simplex terminates if $v(x_1) - u_1$ cannot stay same forever.

For that we prove that same state can not repeat in a simplex run.

Wlog, let us suppose the states of sth and tth iterations of simplex is same and there is no change in $v(x_1) - u_1$ from p to q.

Let r be the largest index column which left and reentered NB at iteration p and q respectively, where $s \leq p < q \leq t$.

Simplex - termination(contd.)

Now Consider,

$$\begin{bmatrix} 1 & \underbrace{0}_{m-1} & \mu^p \end{bmatrix} \begin{bmatrix} A^p \\ \mathtt{I} \\ -\mathtt{I} \end{bmatrix} y^q = [-1 \;\; 0] y^q > 0$$

Now we will show that the above term cannot be > 0.

Let us apply a different calculation on the above term.

$$\begin{bmatrix} 1 & \underbrace{0}_{m-1} & \mu^p \end{bmatrix} \begin{bmatrix} A^p \\ \mathtt{I} \\ -\mathtt{I} \end{bmatrix} y^q = \begin{bmatrix} 1 & \underbrace{0}_{m-1} & \mu^p \end{bmatrix} \begin{bmatrix} A^p y^q \\ \mathtt{I} y^q \\ -\mathtt{I} y^q \end{bmatrix}$$

$$= \begin{bmatrix} 1 & \underbrace{0}_{m-1} & \mu^p \end{bmatrix} \begin{bmatrix} 0 \\ \mathtt{I} y^q \\ -\mathtt{I} y^q \end{bmatrix} = \mu^p \begin{bmatrix} y^q \\ -y^q \end{bmatrix}$$

Termination

Let $\hat{y}^q \triangleq \begin{bmatrix} y^q \\ -y^q \end{bmatrix}$

Now we show every $\mu_j \hat{y}_j^p$ is non-positive.

- $j \in B^p$ or $j - n \in B^p$ or jth bound is inactive, $\mu_j^p = 0$
- $j \in NB^p$ or $j - n \in NB^p$, and jth bound is active
  - $j > r$, $y_i^q = 0$
  - $j = r$, $\underbrace{u_r^p < 0}_{\text{because } r \text{ is selected to leave } NB^p}$ and $y_r^q > 0(why?)$
  - $j > r$, $\overbrace{u_j^p \geq 0}$ and $y_j^q \leq 0$ (why?)

End of Lecture 16