

# Optimisation and Operations Research

## Lecture 7: The Simplex Algorithm (Phase I)

Matthew Roughan

`<matthew.roughan@adelaide.edu.au>`

`http:`

`//www.maths.adelaide.edu.au/matthew.roughan/notes/OORII/`

School of Mathematical Sciences,  
University of Adelaide

July 30, 2019

# Section 1

## Simplex Phase I

## Problem Recap

We will start with a problem in *standard equality form*.

$$\operatorname{argmax}_{\mathbf{x}} \left\{ z = \mathbf{c}^T \mathbf{x} + z_0 \mid A\mathbf{x} = \mathbf{b}, \mathbf{x} \geq 0 \right\}$$

or in longer form, find  $\mathbf{x}$  that solves

$$\begin{aligned} \max \quad z &= \mathbf{c}^T \mathbf{x} + z_0 \\ \text{such that} \quad A\mathbf{x} &= \mathbf{b} \\ \text{and} \quad \mathbf{x} &\geq 0 \end{aligned}$$

However, the problem might not start in feasible canonical form:

- could be **infeasible**, e.g.,  $b_i < 0$
- could be **“uncanonical,”** e.g., lacks one or more unit columns

# The Simplex Tableau for Phase I Example

Example (uncanonical and infeasible standard equality form)

$$\begin{array}{rcl}
 \max z & = & 2x_2 + x_3 + 6x_4 + 2 \\
 \text{such that} & & x_1 - x_2 \phantom{+ 2x_3} - 2x_4 \phantom{+ x_5} = 0 \\
 & & x_1 + x_2 + 2x_3 \phantom{- 2x_4} = 4 \\
 & & \phantom{x_1 +} x_2 \phantom{+ 2x_3} - x_4 - x_5 = -2 \\
 & & x_i \geq 0, \forall i
 \end{array}$$

becomes the Tableau

$x_1$	$x_2$	$x_3$	$x_4$	$x_5$	$z$	$b$
1	-1	0	-2	0	0	0
1	1	2	0	0	0	4
0	1	0	-1	-1	0	-2
0	-2	-1	-6	0	1	2

$z$  row

There are no basic columns, and one of the  $b_i < 0$

# Intuition for Phase I

- Construct an *artificial problem*, which we specifically create so that
  - ▶ it has a *guaranteed* feasible starting point
  - ▶ maximising its objective *creates a feasible solution* to the original problem
- Use Simplex Phase II on the artificial problem  
Each step (roughly)
  - ▶ adds a basic variable to the original problem
  - ▶ moves towards a feasible solution
- When we stop, we know we will either
  - ▶ have a feasible canonical tableau
  - ▶ have proved that the original problem was infeasible

# Relaxation

- *Relaxation* means giving up on one or more constraints
- Here we want to solve for  $A\mathbf{x} = \mathbf{b}$ , but we could relax this to the constraint  $A\mathbf{x} \leq \mathbf{b}$ .
- The idea is that
  - ▶ we know how to put this in a suitable form to solve: just add *artificial* slack variables  $a_j$  to get

$$A\mathbf{x} + \mathbf{a} = \mathbf{b}$$

- ▶ artificial slack variables measure how far away we are from feasibility

# Creating an Artificial Problem via Relaxation

- 1 Multiply any row with a  $b_i < 0$  by  $-1$
- 2 Relax  $A\mathbf{x} = \mathbf{b} \rightarrow A\mathbf{x} \leq \mathbf{b}$ .
- 3 Add in *artificial* slack variables  $a_i$  to get

$$A\mathbf{x} + \mathbf{a} = \mathbf{b}$$

The relaxed problem is *guaranteed* to be in feasible canonical form

- it is feasible, because we ensured that all  $b_i \geq 0$  before we started
- it is canonical because we can write it as

$$B\mathbf{y} = \mathbf{b}$$

where  $B = [A \mid I_m]$  and  $\mathbf{y}^T = (\mathbf{x}^T, \mathbf{a}^T)$ .

We still need an artificial objective function.

## How do we measure infeasibility?

We don't (for the moment) care about the original objective, we instead create an artificial objective to pull us back to the original equations

- the artificial variables  $a_i$  measure how far we are from feasible
- maximise new objective

$$u = - \sum_{i=1}^m a_i = \sum_{i=1}^m [A\mathbf{x} - \mathbf{b}]_i$$

- we created the  $a_i$  such that  $a_i \geq 0$ , so  $u \leq 0$
- when  $u = 0$  we have  $A\mathbf{x} = \mathbf{b}$ , *i.e.*, feasibility
- So we want to maximise  $u$ , with the aim of getting 0



## How do we build this objective into the problem?

$$\begin{aligned}u &= \sum_{i=1}^m \left( \sum_{j=1}^n a_{ij} x_j - b_i \right) \\&= \sum_{i=1}^m \sum_{j=1}^n a_{ij} x_j - \sum_{i=1}^m b_i \\&= \sum_{j=1}^n x_j \sum_{i=1}^m a_{ij} - \sum_{i=1}^m b_i \\&= \sum_{j=1}^n x_j d_j + u_0\end{aligned}$$

where we define

$$d_j = \sum_{i=1}^m a_{ij} \quad \text{and} \quad u_0 = - \sum_{i=1}^m b_i$$

# Artificial problem

So

- We multiply any constraint row with  $b_i < 0$  by  $-1$
- We introduce artificial variables  $a_i$  so that the system is in feasible canonical form
- We maximise  $u = \sum_{i=1}^n x_j d_j + u_0$
- We also note that

$$z = \mathbf{c}^T \mathbf{x} + z_0$$

which we can rewrite as a constraint

$$-\mathbf{c}^T \mathbf{x} + z = z_0$$

where  $z$  looks like another variable

- ▶ we don't know the optimal value of  $z$  yet, its just another variable for the purpose of this optimisation.

## Now we have a new LP

$$\begin{array}{ll} \max & u = \mathbf{d}^T \mathbf{x} + u_0 \\ \text{such that} & B\mathbf{y} = \mathbf{b} \\ & -\mathbf{c}^T \mathbf{x} + z = z_0 \\ \text{and} & \mathbf{y} \geq 0 \end{array}$$

- This is a standard equality form LP with  $n + 1$  variables and  $m + 1$  constraints
  - ▶ it starts in feasible canonical form
  - ▶ so we can solve it using the Simplex Phase II algorithm
- But there are some differences
  - 1 we don't pivot in the  $z$  row (see below)
  - 2 the final tableau might have some elements with  $b_i = 0$ , on which we perform a special step
  - 3 unboundedness of this problem = infeasibility of the original problem
- Once we finish, remove the  $u$  row, and continue with Phase II

# The Simplex Tableau for Phase I

The Simplex Phase I Tableau  $M$  for

$$\operatorname{argmax}_{\mathbf{x}} \left\{ \mathbf{u} = \mathbf{d}^T \mathbf{x} + u_0 \mid \mathbf{A}\mathbf{x} = \mathbf{b}, -\mathbf{c}^T \mathbf{x} + z = z_0, \mathbf{x} \geq 0 \right\}$$

is

	$a_j$ 's	$z$ col.	$u$ col.	$b$ col.	
$M =$	$A$	$I$	$\mathbf{0}$	$\mathbf{0}$	$\mathbf{b}$
	$-\mathbf{c}^T$	$\mathbf{0}$	1	0	$z_0$ $z$ row
	$-\mathbf{d}^T$	$\mathbf{0}$	0	1	$u_0$ $u$ row

- Pre-multiply any row with a negative  $b_i$  by -1 before construction
- $d_j = \sum_{i=1}^m a_{ij}$
- Start with artificial variables as basic
- We don't really need to keep track of the artificial variables because they will be zero in the end anyway

# The Simplex Tableau for Phase I Example

## Example

$$\begin{aligned}
 \max z &= 2x_2 + x_3 + 6x_4 + 2 \\
 \text{such that} & \quad x_1 - x_2 - 2x_4 = 0 \\
 & \quad x_1 + x_2 + 2x_3 = 4 \\
 & \quad x_2 - x_4 - x_5 = -2 \\
 & \quad x_i \geq 0, \forall i
 \end{aligned}$$

becomes the Tableau

$x_1$	$x_2$	$x_3$	$x_4$	$x_5$	$a_1$	$a_2$	$a_3$	$z$	$u$	$b$	
1	-1	0	-2	0	1	0	0	0	0	0	
1	1	2	0	0	0	1	0	0	0	4	
0	-1	0	1	1	0	0	1	0	0	2	$\times -1$
0	-2	-1	-6	0	0	0	0	1	0	2	$z$ row
-2	1	-2	1	-1	0	0	0	0	1	-6	$u$ row

In calculations, we can drop the artificial  $a_i$  and  $u$  columns (as we could with the  $z$  column earlier).

# The Simplex Phase I Pivot Selection

- Simplex Phase I then proceeds using the Phase II algorithm
  - ▶ series of pivots
- However, pivot selection is slightly different
  - ▶ don't pivot in the z-row
  - ▶ we might use different discretionary rules

# The Simplex Phase I Termination

- Simplex Phase I then proceeds using the Phase II algorithm
  - ▶ series of pivots
  - ▶ terminates when we can't find a new pivot
- The artificial problem
  - ▶ is guaranteed to be feasible (we set it up that way)
  - ▶ is guaranteed to be bounded because we know  $u \leq 0$

Hence we will find an optimal solution  $u^*$  to the artificial problem

- There are two cases
  - ▶  $u^* < 0 \Rightarrow$  the original problem is infeasible
  - ▶  $u^* = 0 \Rightarrow$  the original problem is feasible, and the solution to the artificial problem (minus the artificial variables) is a feasible starting point for the original problem
- Its possible we won't have all the required unit columns
  - ▶ we can multiply rows with  $b_i = 0$  by -1 to get them

# Simplex Phase I pseudo-code

Problem in standard equality form:  $\max_{\mathbf{x}} \{z = \mathbf{c}^T \mathbf{x} + z_0 \mid \mathbf{A}\mathbf{x} = \mathbf{b}, \mathbf{x} \geq 0\}$

```
Input:  $A, \mathbf{b}, \mathbf{c}, z_0$  // implicitly  $n$  variables,  $m$  constraints  
Output:  $\mathbf{x}^*, z^*$   
1 Construct Simplex Phase I Tableau  $M$   
2 while at least one  $-d_i < 0$  do  
3 |   Select Entering Variable  $x_j$  (column  $j$ )  
4 |   Select Leaving Variable  $x_k$  (row  $i$ )  
5 |    $\ell_B(i) \leftarrow j$   
6 |    $M \leftarrow \text{Pivot}(M, i, j)$   
7 end  
8  $u^* \leftarrow M(\text{u-row}, \text{b-col})$   
9 if  $u^* < 0$  then  
10 |   problem is infeasible  
11 else  
12 |   remove any all-zero rows  
13 |   for row  $i$  without a basic column do  
14 | |    $M \leftarrow \text{Pivot}(M, i, j)$ , s.t.  $a_{ij} \neq 0$  //  $b_i = 0$  for such rows  
15 |   end  
16 |   remove  $u$ -row and col, and artificial variables  
17 |   continue to Simplex Phase II  
18 end
```

**Algorithm 1:** Simplex Phase I



# Simplex Method Phase I Example: iteration 1

## Example

$x_1$	$x_2$	$x_3$	$x_4$	$x_5$	$z$	$u$	$b$	
1	-1	0	-2	0	0	0	0	
1	1	2	0	0	0	0	4	
0	-1	0	1	1	0	0	2	
0	-2	-1	-6	0	1	0	2	$z$ row
-2	1	-2	1	-1	0	1	-6	$u$ row

⇓

1	-1	0	-2	0	0	0	0	
0	2	2	2	0	0	0	4	
0	-1	0	1	1	0	0	2	
0	-2	-1	-6	0	1	0	2	$z$ row
0	-1	-2	-3	-1	0	1	-6	$u$ row

# Simplex Method Phase I Example: iteration 2

## Example

$x_1$	$x_2$	$x_3$	$x_4$	$x_5$	$z$	$u$	$b$	
1	-1	0	-2	0	0	0	0	
0	2	2	2	0	0	0	4	
0	-1	0	1	1	0	0	2	
0	-2	-1	-6	0	1	0	2	$z$ row
0	-1	-2	-3	-1	0	1	-6	$u$ row

↓

1	0	1	-1	0	0	0	2	
0	1	1	1	0	0	0	2	
0	0	1	2	1	0	0	4	
0	0	1	-4	0	1	0	6	$z$ row
0	0	-1	-2	-1	0	1	-4	$u$ row

# Simplex Method Phase I Example: iteration 3

## Example

1	0	1	-1	0	0	0	2	
0	1	1	1	0	0	0	2	
0	0	1	2	1	0	0	4	
0	0	1	-4	0	1	0	6	<i>z</i> row
0	0	-1	-2	-1	0	1	-4	<i>u</i> row



1	0	3/2	0	1/2	0	0	4	
0	1	1/2	0	-1/2	0	0	0	
0	0	1/2	1	1/2	0	0	2	
0	0	3	0	2	1	0	14	<i>z</i> row
0	0	0	0	0	0	1	0	<i>u</i> row

# Simplex Method Phase I Example: stop

## Example

$x_1$	$x_2$	$x_3$	$x_4$	$x_5$	$z$	$u$	$b$	
1	0	3/2	0	1/2	0	0	4	
0	1	1/2	0	-1/2	0	0	0	
0	0	1/2	1	1/2	0	0	2	
0	0	3	0	2	1	0	14	$z$ row
0	0	0	0	0	0	1	0	$u$ row

- There are no more places with  $-d_j < 0$ , so Simplex Phase I stops
- The Tableau is in feasible canonical form

$$\ell_B = (1, 2, 4)$$

- Also  $u^* = 0$ , so we have found a feasible starting point for Phase II
- Drop the  $u$ -row and proceed to Phase II (but that will finish straight away because the resulting Tableau is already finished)

# Phase I Properties

How does Phase I guarantee to get us to a feasible canonical form?

- We never pivot on the columns corresponding to the  $a_i$ 
  - ▶ in fact, we can leave these columns out of calculations
  - ▶ so these are never entering variables
- Take  $R$  to be the set of rows without a corresponding basic variable (outside the artificial variables)
  - ▶ if we ever pivot in row  $i$ , this row is removed from  $R$  and never returns
- We know we can always select a new row (as long as we have a column), because the problem is bounded
- When we finish, if  $u_0 = 0$ , then either
  - ▶ we did a pivot in every row, and  $R = \phi$
  - ▶ or we need to think a little more ...

# Simplex Method Phase I Properties

Row  $k$  of  $M$  is denoted  $\mathbf{r}_k$ , but write  $\mathbf{u}$  for the special  $u$ -row.

## Theorem

For each tableau in Phase I

$$\mathbf{u} = - \sum_{i \in R} \mathbf{r}_i, \text{ i.e., } d_j = \sum_{k \in R} a_{kj}, \text{ and } u_0 = - \sum_{k \in R} b_k$$

**Proof :** (by induction):

The equation is true for the initial tableau by construction.

Let's suppose that the result is true for some tableau  $M$

Perform a pivot at  $(i, j)$  (satisfying Simplex pivot rules) to go from

$$M \rightarrow M'$$

then proof by induction requires us to show the result is true for  $M'$ .

# Simplex Method Phase I Properties

Perform a pivot at  $(i, j)$  to go from  $M \rightarrow M'$

$$\begin{aligned} \mathbf{u}' &= \mathbf{u} + \frac{d_j}{a_{ij}} \mathbf{r}_i \\ &= - \sum_{k \in R} \mathbf{r}_k + \frac{\sum_{k \in R} a_{kj} \mathbf{r}_i}{a_{ij}} \\ &= - \sum_{k \in R, k \neq i} \left( \mathbf{r}_k - \frac{a_{kj}}{a_{ij}} \mathbf{r}_i \right) - \left( \mathbf{r}_i - \frac{a_{ij}}{a_{ij}} \mathbf{r}_i \right) \\ &= - \sum_{k \in R, k \neq i} \left( \mathbf{r}_k - \frac{a_{kj}}{a_{ij}} \mathbf{r}_i \right) \\ &= - \sum_{k \in R'} \left( \mathbf{r}_k - \frac{a_{kj}}{a_{ij}} \mathbf{r}_i \right) \end{aligned}$$

Q.E.D.

# Simplex Method, Phase I Properties

Now consider the case  $u_0 = 0$  but  $R$  is non-empty

- we have reached the end
- but haven't got a complete set of basic columns

Then

- the solution must be feasible  $b_i \geq 0$
- from previous theorem  $\sum_{i \in R} b_i = -u_0 = 0$
- so  $b_i = 0$  for all  $i \in R$
- if all elements in the row are equal to 0 then this row is a linear combination of the other rows and is therefore redundant and can be removed.
- otherwise we can pivot on any element  $a_{ij} \neq 0$  in such a row without changing the last column of the tableau
  - ▶ we do these pivots to create our necessary basic variables



## Section 2

# Putting it together

# Simplex Method Altogether

The Simplex Algorithm (Phase I and II) always stops in one of the following ways:

- *stop 1*: (at end of in Phase 1) there are no feasible solutions.
  - ▶ because  $u^* < 0$
- *stop 2*: (during Phase 2) there are feasible solutions with arbitrarily large  $z$ -values, and therefore no optimal solutions;
  - ▶ because we couldn't find a valid pivot row
- *stop 3*: (at the end of Phase 2) feasible optimal solution has been found;
  - ▶ because we couldn't find a valid pivot column

## Mix and Match

Sometimes we start with a problem that has inequalities, and equalities. We could just convert to standard form, and continue, but the initial inequalities already introduce slack variables (and hence basic columns), so we don't really need artificial variables for these, so for

$$\operatorname{argmax}_{\mathbf{x}} \left\{ z = \mathbf{c}^T \mathbf{x} + z_0 \mid B\mathbf{x} \leq \mathbf{b}, D\mathbf{x} = \mathbf{d}, \mathbf{x} \geq 0 \right\}$$

the Simplex Phase I Tableau would be

	$s$ 's	$a$ 's	$z$ col.	$u$ col.	$b$ col.	
$M =$	$B$	$I$	$0$	$0$	$0$	$\mathbf{b}$
	$D$	$0$	$I$	$0$	$0$	$\mathbf{d}$
	$-\mathbf{c}^T$	$0$	$0$	$1$	$0$	$z_0$ $z$ row
	$-\mathbf{d}^T$	$0$	$0$	$0$	$1$	$u_0$ $u$ row

# Mix and Match Example

## Example

$$\begin{array}{rclclclcl} \max z = & & x_2 & & & - & 3x_4 & + & 2 \\ & x_1 & & + & x_3 & - & x_4 & = & 2 \\ & & x_2 & + & x_3 & + & x_4 & = & 2 \\ & & x_2 & & & - & x_4 & \leq & 2 \\ & & & & & & & x_i & \geq & 0 \end{array}$$

The Phase I Tableau is

$x_1$	$x_2$	$x_3$	$x_4$	$x_5$	$z$	$u$	<b>b</b>
1	0	1	-1	0	0	0	2
0	1	1	1	0	0	0	2
0	1	0	-1	1	0	0	2
0	1	0	-3	0	1	0	2
-1	-2	-2	1	1	0	1	-6

# Takeaway

## Why do we teach Simplex?

- because it shows one of the greatest ideas in algorithm development
  - ▶ construct a new problem that is simpler to solve, but helps you with your original problem
- this idea is reused in solution of many mathematical problems

# Further reading I