ORF 307: Lecture 16

Linear Programming: Chapter 23: Integer Programming

Robert J. Vanderbei

April 16, 2019

Slides last edited on April 17, 2019

Airline Equipment Scheduling

Given:

- A set of *flight legs* (e.g. Newark to Chicago departing 7:45am).
- A set of aircraft.

Problem: which specific aircraft should fly which flight legs?

Model:

- Generate a set of feasible *routes* (i.e., a collection of legs which taken together can be flown by one airplane).
- Assign a cost to each route (e.g. 1).
- Pick a minimum cost collection of routes that exactly covers all of the legs.

Let:

$$
\begin{aligned}
x_j &= \begin{cases} 1 & \text{if route } j \text{ is selected,} \\ 0 & \text{otherwise} \end{cases} \\
a_{ij} &= \begin{cases} 1 & \text{if leg } i \text{ is part of route } j, \\ 0 & \text{otherwise} \end{cases} \\
c_j &= \text{cost of using route } j.
\end{aligned}
$$

An Integer Programming Problem:

$$
\begin{aligned}
\text{minimize} \quad & \sum_{j=1}^{n} c_j x_j \\
\text{subject to} \quad & \sum_{j=1}^{n} a_{ij} x_j = 1 \qquad & i = 1, 2, \ldots, m, \\
& x_j \in \{0, 1\} \qquad & j = 1, 2, \ldots, n.
\end{aligned}
$$

An example of *set-partitioning problems*.

Airline Crew Scheduling

Similar to equipment scheduling except:

It's possible to put more than one crew on a flight:

- only one crew works
- any others are just being shuttled

Integer Programming Problem:

$$\begin{array}{lll} \text{minimize} & \displaystyle\sum_{j=1}^{n} c_j x_j & \\ \text{subject to} & \displaystyle\sum_{j=1}^{n} a_{ij} x_j \geq 1 & i = 1, 2, \ldots, m, \\ & x_j \in \{0, 1\} & j = 1, 2, \ldots, n. \end{array}$$

An example of *set-covering problems*.

Column Generation

The problem of producing a set of possible routes is called *column generation.*

It is important and nontrivial.

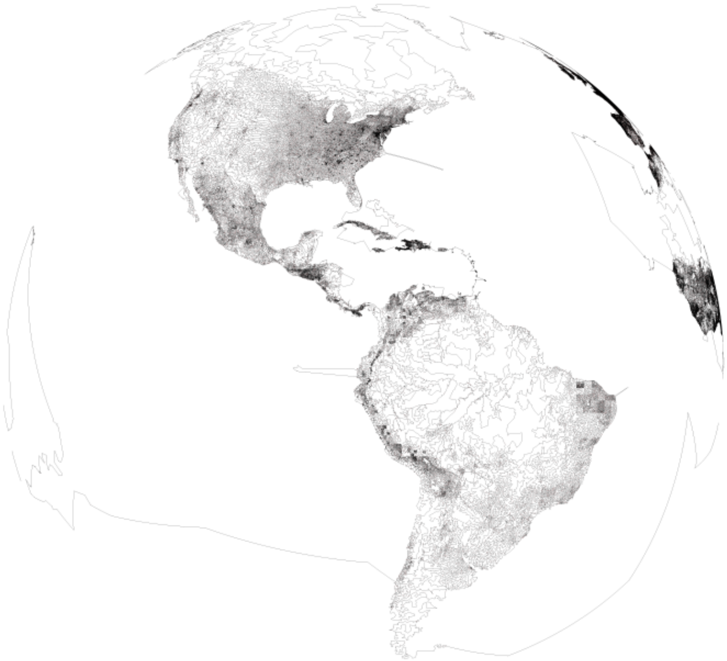
Reason: there are lots of routes.

For example, on a weekly schedule a route might consist of 20 legs.

If there are m legs in total, then there are up to m^{20} possible routes.

Traveling Salesman Problem

Most famous example of a *hard* problem:



Given n cities, determine the order in which to visit them so as to minimize the total travel distance.

Fixed Costs

A jump at $x = 0$:

$$c(x) = \begin{cases} 0 & \text{if } x = 0 \\ K + cx & \text{if } x > 0. \end{cases}$$

where

$$0 \leq x \leq u.$$

Equivalent to:

$$c(x, y) = Ky + cx$$

together with the following constraints:

$$\begin{aligned} x &\leq uy \\ x &\geq 0 \\ y &\in \{0, 1\}. \end{aligned}$$

Nonlinear Objective Functions

Nonlinear objective functions are sometimes approximated by piecewise linear functions.

Piecewise linear functions can be treated using techniques similar to the fixed cost method above.

LP Relaxation

General Integer Programming Problem

$$
\begin{array}{ll}
\text{maximize} & c^T x \\
\text{subject to} & Ax \leq b \\
& x \geq 0 \\
& x \ \text{ has integer components.}
\end{array}
$$

Nonlinear Objective Functions

Nonlinear objective functions are sometimes approximated by piecewise linear functions.

Piecewise linear functions can be treated using techniques similar to the fixed cost method above.

LP Relaxation

General Integer Programming Problem

$$\begin{aligned} \text{maximize} \quad & c^T x \\ \text{subject to} \quad & Ax \le b \\ & x \ge 0 \end{aligned}$$

~~x has integer components.~~

Example

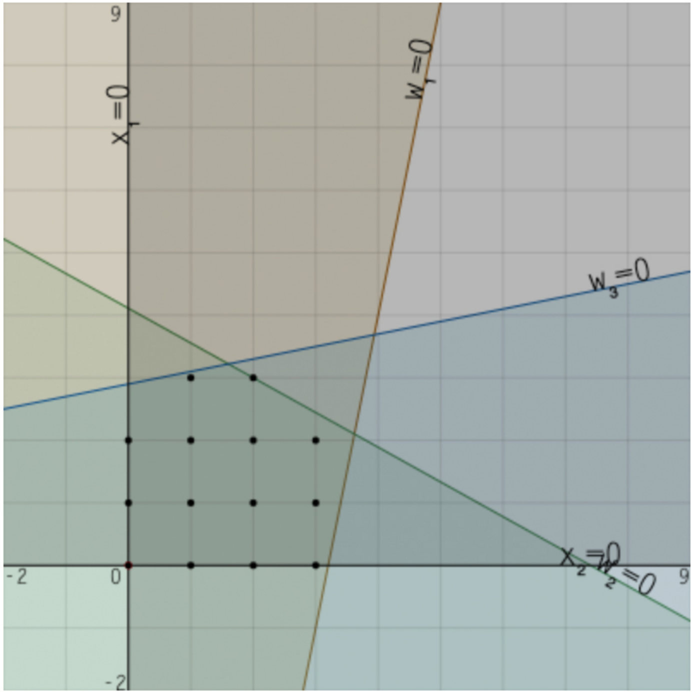
Hint Undo Redo Number format: Fraction

Current Dictionary:

maximize ζ	=	0	+	8	x_1	+	13	x_2
w_1	=	32	-	10	x_1	-	-2	x_2
w_2	=	37	-	5	x_1	-	9	x_2
w_3	=	29	-	-2	x_1	-	10	x_2

$x_1, x_2 \geq 0$ integer, $w_1, w_2, w_3 \geq 0$

Add Cuts Optimal Infeasible Unbounded



Example with Upper/Lower Bounds

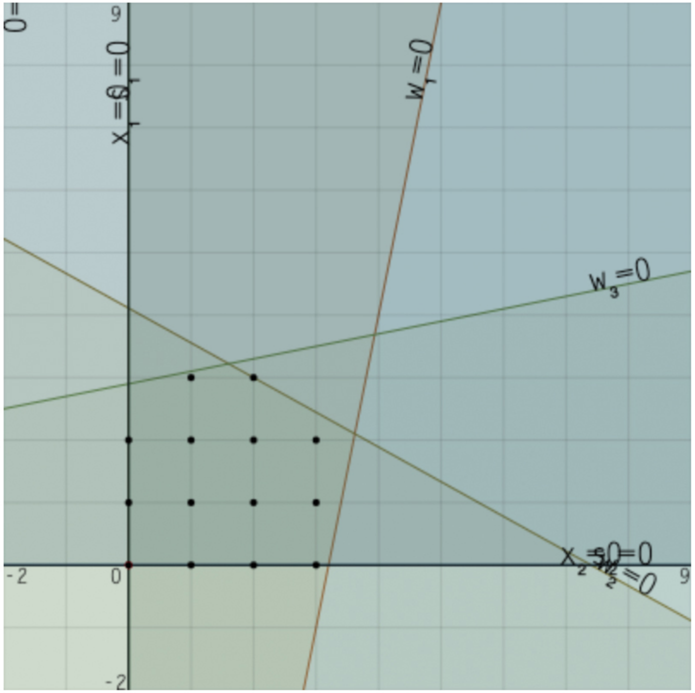
Hint | Undo | Redo | Number format: Fraction

Current Dictionary:

maximize ζ	=	0	+	8	x_1	+	13	x_2
w_1	=	32	-	10	x_1	-	-2	x_2
w_2	=	37	-	5	x_1	-	9	x_2
w_3	=	29	-	-2	x_1	-	10	x_2
s_1	=	0	-	-1	x_1	-	0	x_2
t_1	=	10	-	1	x_1	-	0	x_2
s_2	=	0	-	0	x_1	-	-1	x_2
t_2	=	10	-	0	x_1	-	1	x_2

$x_1, x_2 \geq 0$ integer, $w_1, w_2, w_3, s_1, t_1, s_2, t_2 \geq 0$

Add Cuts | Optimal | Infeasible | Unbounded



For the *Branch and Bound* method, we have introduced upper and lower bound constraints:

$$0 \leq x_1 \leq 10, \qquad 0 \leq x_2 \leq 10.$$

The number 10 is just taken to be some "very large" number (aka infinity). It will get changed to smaller values as we go.

Optimal Solution to LP Relaxation

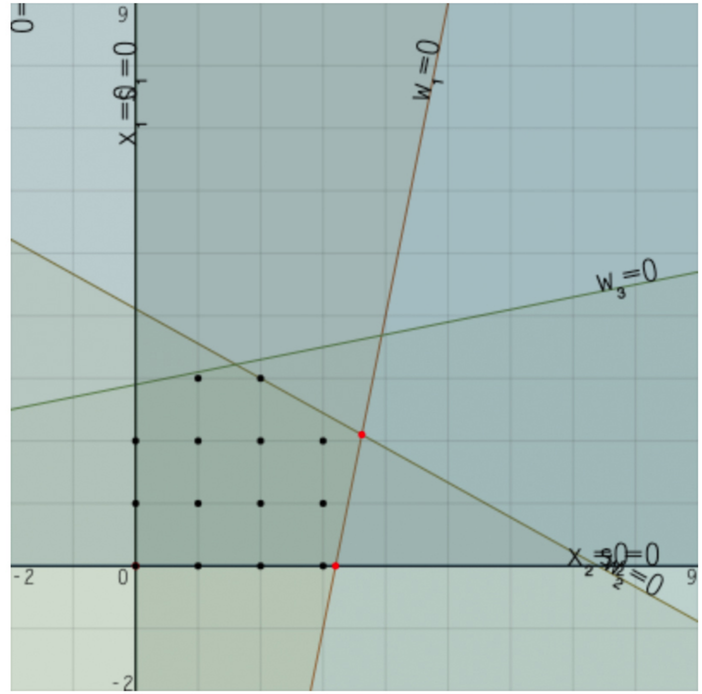
Hint | Undo | Redo | Number format: Fraction

Current Dictionary:

maximize ζ	=	2813/5(	+	-7/100	w_1	+	-73/50	w_2
x_1	=	181/50	-	9/100	w_1	-	1/50	w_2
x_2	=	21/10	-	-1/20	w_1	-	1/10	w_2
w_3	=	381/25	-	17/25	w_1	-	-24/25	w_2
s_1	=	181/50	-	9/100	w_1	-	1/50	w_2
t_1	=	319/50	-	-9/100	w_1	-	-1/50	w_2
s_2	=	21/10	-	-1/20	w_1	-	1/10	w_2
t_2	=	79/10	-	1/20	w_1	-	-1/10	w_2

$x_1, x_2 \geq 0$ integer, $w_1, w_2, w_3, s_1, t_1, s_2, t_2 \geq 0$

Add Cuts | Optimal | Infeasible | Unbounded



Optimal solution is $(x_1, x_2) = (181/50, 21/10) = (3.62, 2.1)$ with objective value $2813/50 = 56.26$.

Rounding to integers: $(4, 2)$ $\Longleftarrow$ infeasible.

Closest feasible: $(3, 2)$ $\Longleftarrow$ suboptimal (as we'll see).

Introduce an *upper bound* of 3 on x_1:

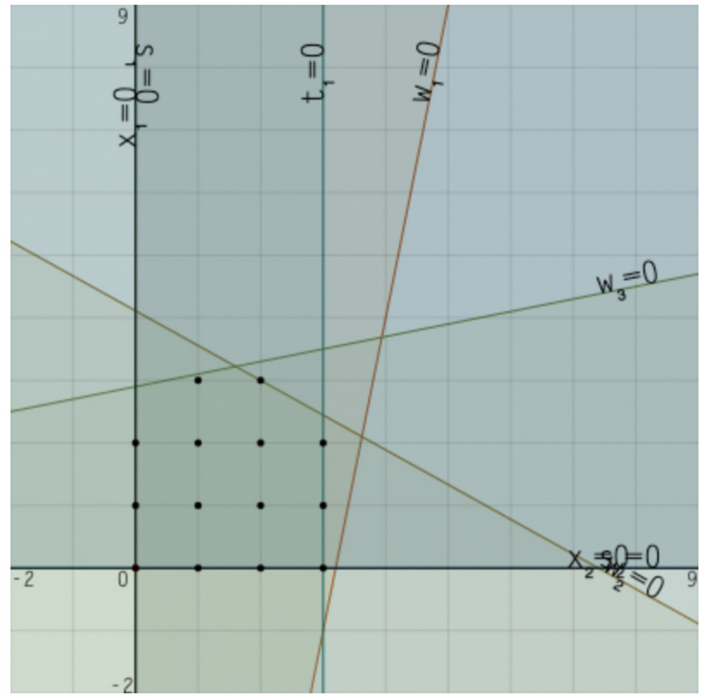
Hint | Undo | Redo | Number format: Fraction

Current Dictionary:

maximize ζ	=	0	+	8	x_1	+	13	x_2
w_1	=	32	-	10	x_1	-	-2	x_2
w_2	=	37	-	5	x_1	-	9	x_2
w_3	=	29	-	-2	x_1	-	10	x_2
s_1	=	0	-	-1	x_1	-	0	x_2
t_1	=	3	-	1	x_1	-	0	x_2
s_2	=	0	-	0	x_1	-	-1	x_2
t_2	=	10	-	0	x_1	-	1	x_2

$x_1, x_2 \geq 0$ integer, $w_1, w_2, w_3, s_1, t_1, s_2, t_2 \geq 0$

Add Cuts | Optimal | Infeasible | Unbounded



Could solve this problem from scratch, or better yet...

Start From Previous Optimal Dictionary

Originally, we had

$$\begin{aligned} t_1 \quad = \quad 10 - x_1 \quad &= \quad 10 - 181/50 + 9/100 w_1 + 1/50 w_2 \\ &= \quad 319/50 + 9/100 w_1 + 1/50 w_2 \end{aligned}$$

We change this too...

$$\begin{aligned} t_1 \quad = \quad 3 - x_1 \quad &= \quad 3 - 181/50 + 9/100 w_1 + 1/50 w_2 \\ &= \quad -\ 31/50 + 9/100 w_1 + 1/50 w_2 \end{aligned}$$

To summarize: upper bound decreases by 7, then the right-hand side for the upper bound's slack variable decreases by 7. Everything else remains the same...

Start From Previous Optimal Dictionary

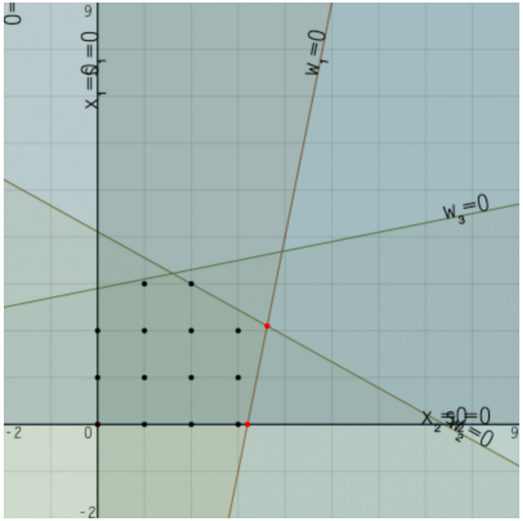
Hint | Undo | Redo | Number format: Fraction

Current Dictionary:

maximize ζ	=	2813/5(	+	-7/100	w_1	+	-73/50 w_2
x_1	=	181/50	-	9/100	w_1	-	1/50 w_2
x_2	=	21/10	-	-1/20	w_1	-	1/10 w_2
w_3	=	381/25	-	17/25	w_1	-	-24/25 w_2
s_1	=	181/50	-	9/100	w_1	-	1/50 w_2
t_1	=	319/50	-	-9/100	w_1	-	-1/50 w_2
s_2	=	21/10	-	-1/20	w_1	-	1/10 w_2
t_2	=	79/10	-	1/20	w_1	-	-1/10 w_2

$x_1, x_2 \geq 0$ integer, $w_1, w_2, w_3, s_1, t_1, s_2, t_2 \geq 0$

Add Cuts | Optimal | Infeasible | Unbounded



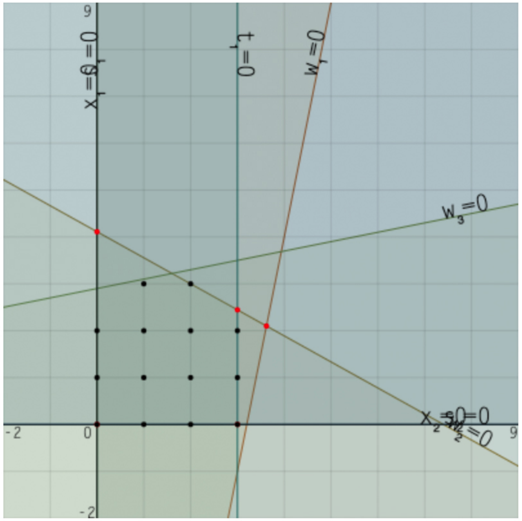
Hint | Undo | Redo | Number format: Fraction

Current Dictionary:

maximize ζ	=	2813/5(	+	-7/100	w_1	+	-73/50 w_2
x_1	=	181/50	-	9/100	w_1	-	1/50 w_2
x_2	=	21/10	-	-1/20	w_1	-	1/10 w_2
w_3	=	381/25	-	17/25	w_1	-	-24/25 w_2
s_1	=	181/50	-	9/100	w_1	-	1/50 w_2
t_1	=	-31/50	-	-9/100	w_1	-	-1/50 w_2
s_2	=	21/10	-	-1/20	w_1	-	1/10 w_2
t_2	=	79/10	-	1/20	w_1	-	-1/10 w_2

$t_1, x_2 \geq 0$ integer, $w_1, w_2, w_3, s_1, x_1, s_2, t_2 \geq 0$

Add Cuts | Optimal | Infeasible | Unbounded



First Branch of Branch&Bound–Optimal Solution

Can be solved with just one dual pivot.

Here's the optimal solution...

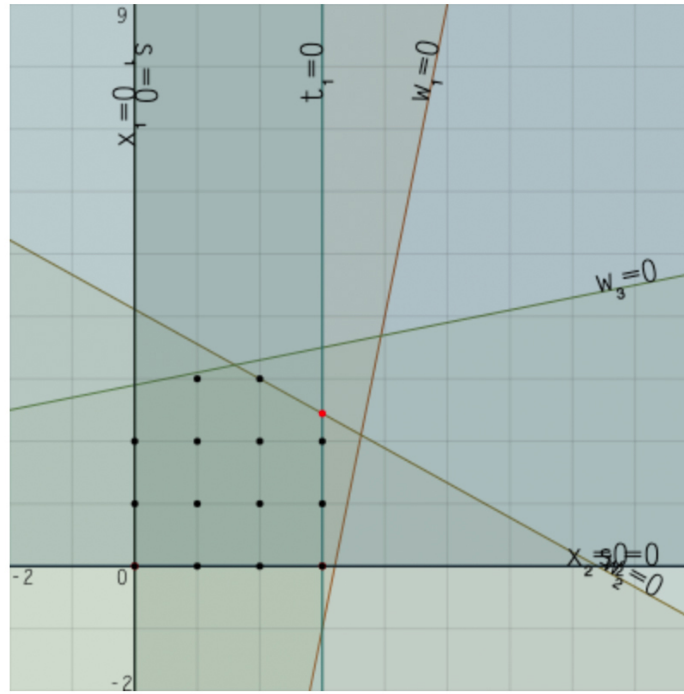
Hint Undo Redo Number format: Fraction

Current Dictionary:

maximize ζ	=	502/9	+	-7/9	t_1	+	-13/9	w_2
w_1	=	62/9	-	-100/9	t_1	-	2/9	w_2
x_2	=	22/9	-	-5/9	t_1	-	1/9	w_2
w_3	=	95/9	-	68/9	t_1	-	-10/9	w_2
s_1	=	3	-	1	t_1	-	0	w_2
x_1	=	3	-	1	t_1	-	0	w_2
s_2	=	22/9	-	-5/9	t_1	-	1/9	w_2
t_2	=	68/9	-	5/9	t_1	-	-1/9	w_2

$t_1, x_2 \geq 0$ integer, $w_1, w_2, w_3, s_1, x_1, s_2, t_2 \geq 0$

Add Cuts Optimal Infeasible Unbounded



x_1 is an integer but x_2 is not. Gotta keep going.

Second Branch

Before working on x_2, let's ask what happens if we introduce a lower bound of 4 on x_1.

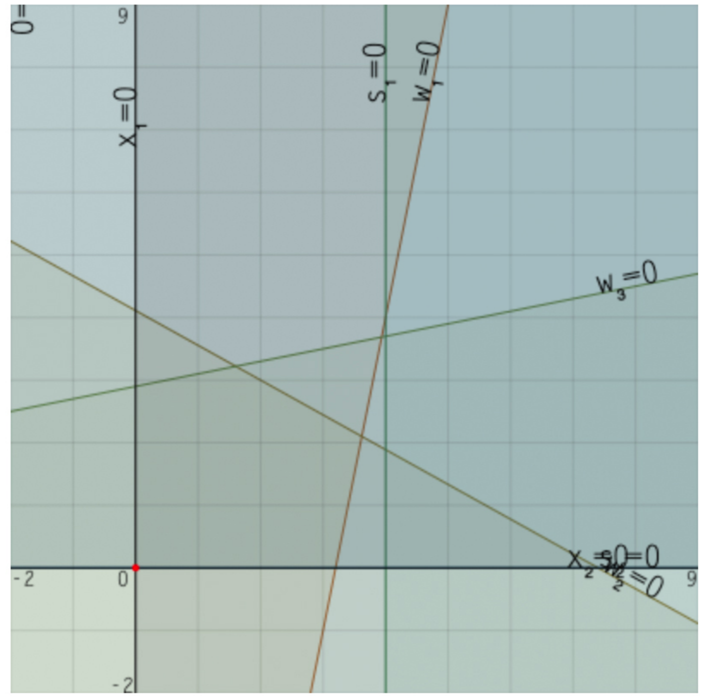
Hint   Undo   Redo   Number format: Fraction

Current Dictionary:

maximize ζ	=	0	+	8	x_1	+	13	x_2
w_1	=	32	-	10	x_1	-	-2	x_2
w_2	=	37	-	5	x_1	-	9	x_2
w_3	=	29	-	-2	x_1	-	10	x_2
s_1	=	-4	-	-1	x_1	-	0	x_2
t_1	=	10	-	1	x_1	-	0	x_2
s_2	=	0	-	0	x_1	-	-1	x_2
t_2	=	10	-	0	x_1	-	1	x_2

$w_1, x_2 \geq 0$ integer, $x_1, w_2, w_3, s_1, t_1, s_2, t_2 \geq 0$

Add Cuts   Optimal   Infeasible   Unbounded



Again, we start from the optimal solution to the original LP relaxation and just update one right-hand side...

First Branch of Branch&Bound–Optimal Solution

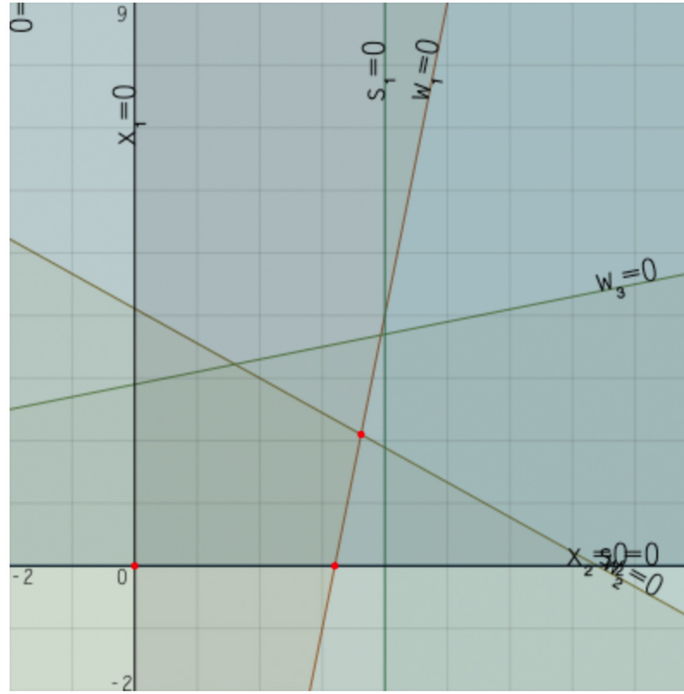
Hint | Undo | Redo | Number format: Fraction

Current Dictionary:

maximize ζ	=	2813/5(	+	-7/100	w_1	+	-73/50	w_2
x_1	=	181/50	-	9/100	w_1	-	1/50	w_2
x_2	=	21/10	-	-1/20	w_1	-	1/10	w_2
w_3	=	381/25	-	17/25	w_1	-	-24/25	w_2
s_1	=	-19/50	-	9/100	w_1	-	1/50	w_2
t_1	=	319/50	-	-9/100	w_1	-	-1/50	w_2
s_2	=	21/10	-	-1/20	w_1	-	1/10	w_2
t_2	=	79/10	-	1/20	w_1	-	-1/10	w_2

$w_1, x_2 \geq 0$ integer, $x_1, w_2, w_3, s_1, t_1, s_2, t_2 \geq 0$

Add Cuts | Optimal | Infeasible | Unbounded



INFEASIBLE!

Let's go back to the scenario where $x_1 \leq 3$ and add an upper bound constraint of 2 on x_2...

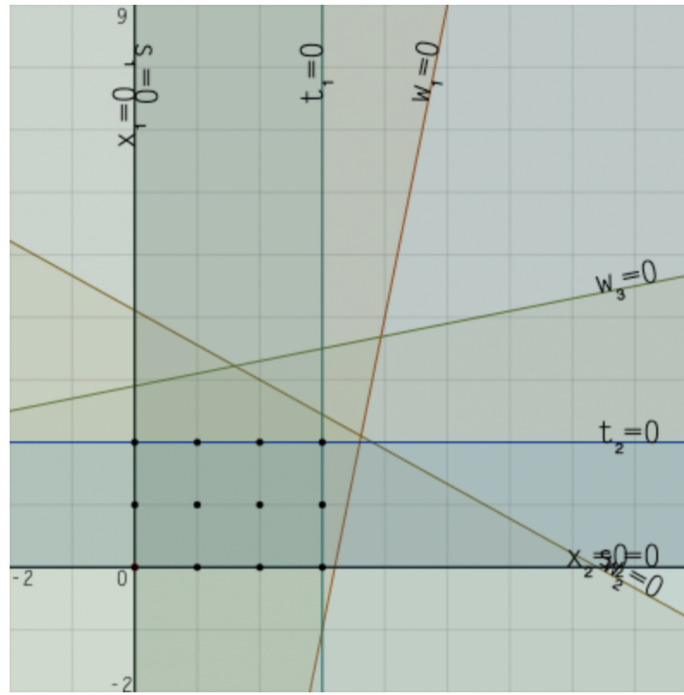
Hint | Undo | Redo | Number format: Fraction

Current Dictionary:

maximize ζ	=	0	+	8	x_1	+	13	x_2
w_1	=	32	-	10	x_1	-	-2	x_2
w_2	=	37	-	5	x_1	-	9	x_2
w_3	=	29	-	-2	x_1	-	10	x_2
s_1	=	0	-	-1	x_1	-	0	x_2
t_1	=	3	-	1	x_1	-	0	x_2
s_2	=	0	-	0	x_1	-	-1	x_2
t_2	=	2	-	0	x_1	-	1	x_2

$x_1, x_2 \geq 0$ integer, $w_1, w_2, w_3, s_1, t_1, s_2, t_2 \geq 0$

Add Cuts | Optimal | Infeasible | Unbounded



Our First All Integer Solution

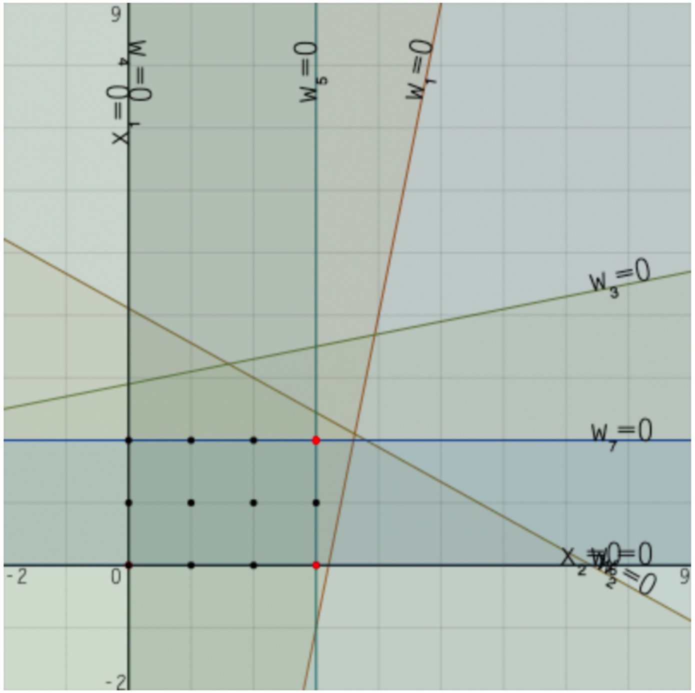
Hint Undo Redo Number format: Fraction

Current Dictionary:

maximize ζ	=	50	+	-8	t_1	+	-13	t_2
w_1	=	6	-	-10	t_1	-	2	t_2
w_2	=	4	-	-5	t_1	-	-9	t_2
w_3	=	15	-	2	t_1	-	-10	t_2
s_1	=	3	-	1	t_1	-	0	t_2
x_1	=	3	-	1	t_1	-	0	t_2
s_2	=	2	-	0	t_1	-	1	t_2
x_2	=	2	-	0	t_1	-	1	t_2

$x_1, x_2 \geq 0$ integer, $w_1, w_2, w_3, w_4, w_5, w_6, w_7 \geq 0$

Add Cuts Optimal Infeasible Unbounded



It's integers, but is it optimal?

Maybe not. Maybe we were wrong to impose the constraint $x_2 \leq 2$.

Maybe it's better to give x_2 a lower bound of 3. Let's see...

A Second Branch Below the First Branch

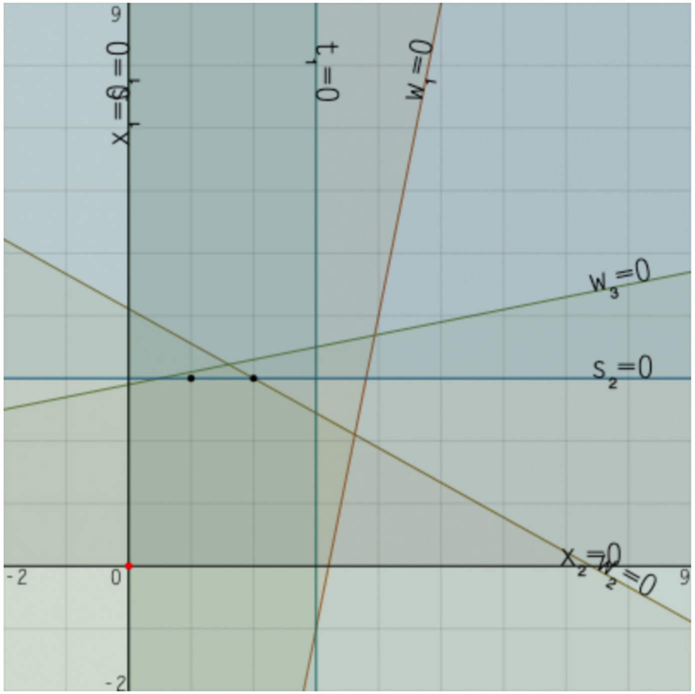
Hint Undo Redo Number format: Fraction

Current Dictionary:

maximize ζ	=	0	+	8	x_1	+	13	x_2	
w_1	=	32	-	10	x_1	-	-2	x_2	
w_2	=	37	-	5	x_1	-	9	x_2	
w_3	=	29	-	-2	x_1	-	10	x_2	
s_1	=	0	-	-1	x_1	-	0	x_2	
t_1	=	3	-	1	x_1	-	0	x_2	
s_2	=	-3	-	0	x_1	-	-1	x_2	
t_2	=	10	-	0	x_1	-	1	x_2	

$x_1, x_2 \geq 0$ integer, $w_1, w_2, w_3, s_1, t_1, s_2, t_2 \geq 0$

Add Cuts Optimal Infeasible Unbounded



Another All Integer Solution

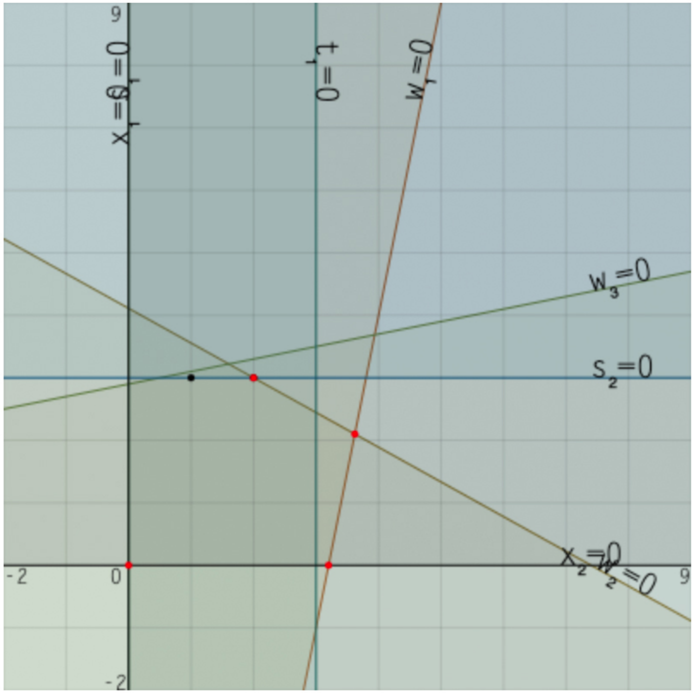
Hint | Undo | Redo | Number format: Fraction

Current Dictionary:

maximize ζ =	55	+	-7/5 s_2	+	-8/5 w_2
x_1 =	2	-	9/5 s_2	-	1/5 w_2
x_2 =	3	-	-1 s_2	-	0 w_2
w_3 =	3	-	68/5 s_2	-	2/5 w_2
s_1 =	2	-	9/5 s_2	-	1/5 w_2
t_1 =	1	-	-9/5 s_2	-	-1/5 w_2
w_1 =	18	-	-20 s_2	-	-2 w_2
t_2 =	7	-	1 s_2	-	0 w_2

$x_1, x_2 \geq 0$ integer, $w_1, w_2, w_3, s_1, t_1, s_2, t_2 \geq 0$

Add Cuts | Optimal | Infeasible | Unbounded



And, look at the objective function value.

This one's better than the one we had before.

There's no more branching to do. We are done. This is the OPTIMAL SOLUTION!!

Here's the same problem again:

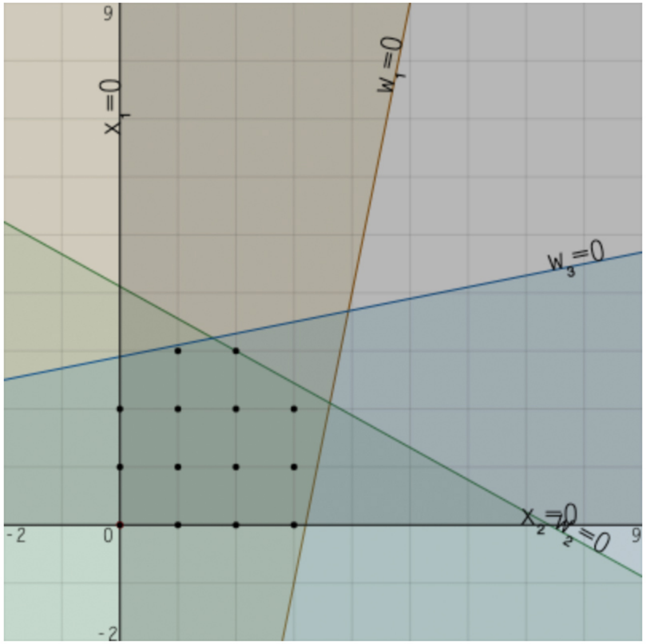
Hint Undo Redo Number format: Fraction

Current Dictionary:

maximize ζ	=	0	+	8	x_1	+	13	x_2
w_1	=	32	-	10	x_1	-	-2	x_2
w_2	=	37	-	5	x_1	-	9	x_2
w_3	=	29	-	-2	x_1	-	10	x_2

$x_1, x_2 \geq 0$ integer, $w_1, w_2, w_3 \geq 0$

Add Cuts Optimal Infeasible Unbounded



NOTE: Seed value 6 in Gomory pivot tool.

LP Relaxation

Here's the solution to the *LP relaxation*:

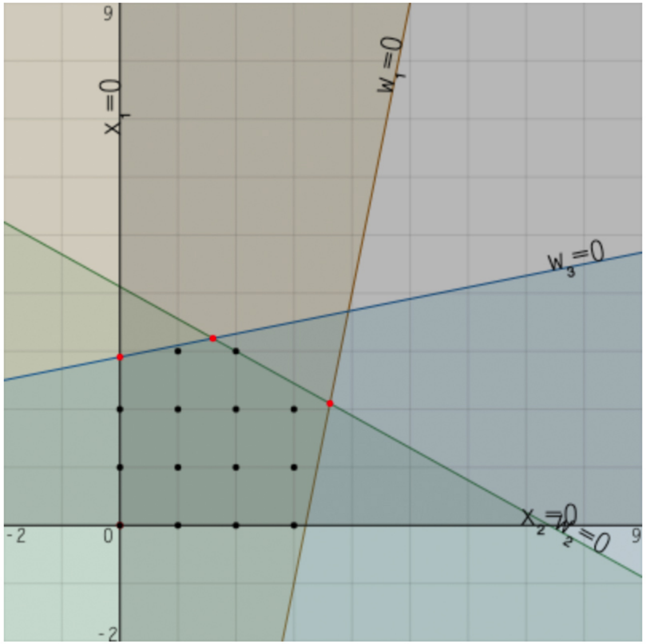
Hint | Undo | Redo | Number format: Fraction

Current Dictionary:

maximize ζ	=	2813/50	+	-73/50	w_2	+	-7/100	w_1
w_3	=	381/25	-	-24/25	w_2	-	17/25	w_1
x_1	=	181/50	-	1/50	w_2	-	9/100	w_1
x_2	=	21/10	-	1/10	w_2	-	-1/20	w_1

$x_1, x_2 \geq 0$ integer, $w_1, w_2, w_3 \geq 0$

Add Cuts | Optimal | Infeasible | Unbounded



Neither x_1 nor x_2 are integers!

Gomory's Idea

Let's focus on x_1. It's a basic variable in the "optimal" dictionary:

$$x_1 = \frac{181}{50} - \frac{1}{50}w_2 - \frac{9}{100}w_1.$$

Let's bring the nonbasic variables over to the left-hand side:

$$x_1 + \frac{1}{50}w_2 + \frac{9}{100}w_1 = \frac{181}{50}.$$

Now, if we round down each of the coefficients on the left to the nearest smaller integer, then the left hand side will be smaller than it was. It will also be an integer whenever the variables are integer and so it will be smaller than the rounded-down value of the right-hand side:

$$x_1 + 0w_2 + 0w_1 \leq 3.$$

We want to add this as a new constraint. It will have a new slack variable, which will be a basic variable. Hence, we want to get rid of x_1 from this constraint. To do that, substitute the equation above that defines x_1 in terms of the nonbasic variables:

$$\frac{181}{50} - \frac{1}{50}w_2 - \frac{9}{100}w_1 + 0w_2 + 0w_1 \leq 3.$$

Hint Undo Redo Number format: Fraction

Current Dictionary:

$$\text{maximize } \zeta = 2813/50 + (-73/50)\, w_2 + (-7/100)\, w_1$$

$$w_3 = 381/25 - (-24/25)\, w_2 - 17/25\, w_1$$

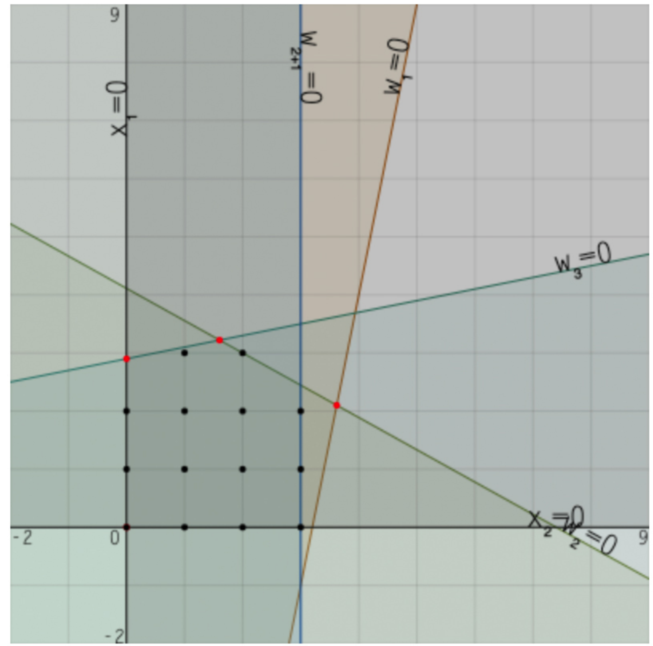
$$x_1 = 181/50 - 1/50\, w_2 - 9/100\, w_1$$

$$x_2 = 21/10 - 1/10\, w_2 - (-1/20)\, w_1$$

$$w_{2+1} = -31/50 - (-1/50)\, w_2 - (-9/100)\, w_1$$

$x_1,\ x_2 \geq 0$ integer, $w_1,\ w_2,\ w_3,\ w_{2+1} \geq 0$

Add Cuts Optimal Infeasible Unbounded



Now we just do a dual pivot to get to a new "optimal" solution:

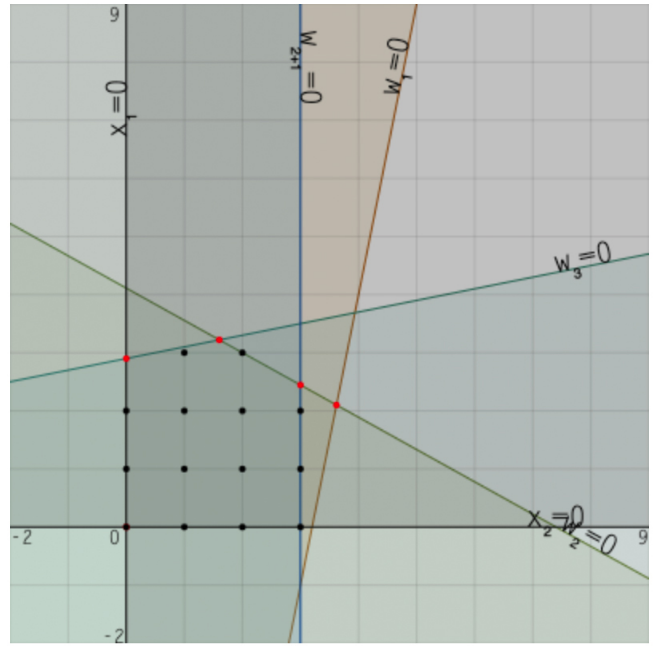
Hint | Undo | Redo | Number format: Fraction

Current Dictionary:

maximize ζ =	502/9	+	-13/9	w_2	+	-7/9	w_{2+1}
w_3 =	95/9	-	-10/9	w_2	-	68/9	w_{2+1}
x_1 =	3	-	0	w_2	-	1	w_{2+1}
x_2 =	22/9	-	1/9	w_2	-	-5/9	w_{2+1}
w_1 =	62/9	-	2/9	w_2	-	-100/9	w_{2+1}

$x_1, x_2 \geq 0$ integer, $w_1, w_2, w_3, w_{2+1} \geq 0$

Add Cuts | Optimal | Infeasible | Unbounded



We still have a non-integer value: $x_2 = 22/9$.

Let's make another Gomory cut:

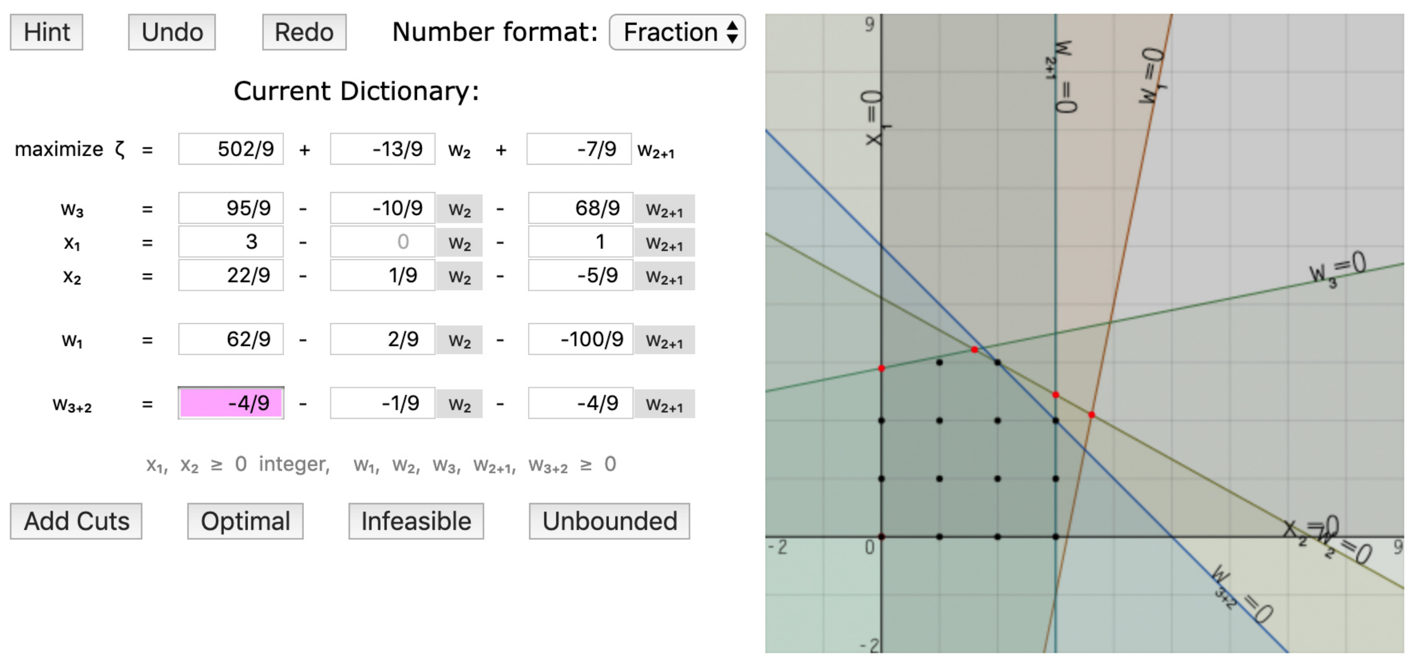
$$x_2 + \frac{1}{9}w_2 - \frac{5}{9}w_{2+1} = \frac{22}{9}.$$

Rounding down...

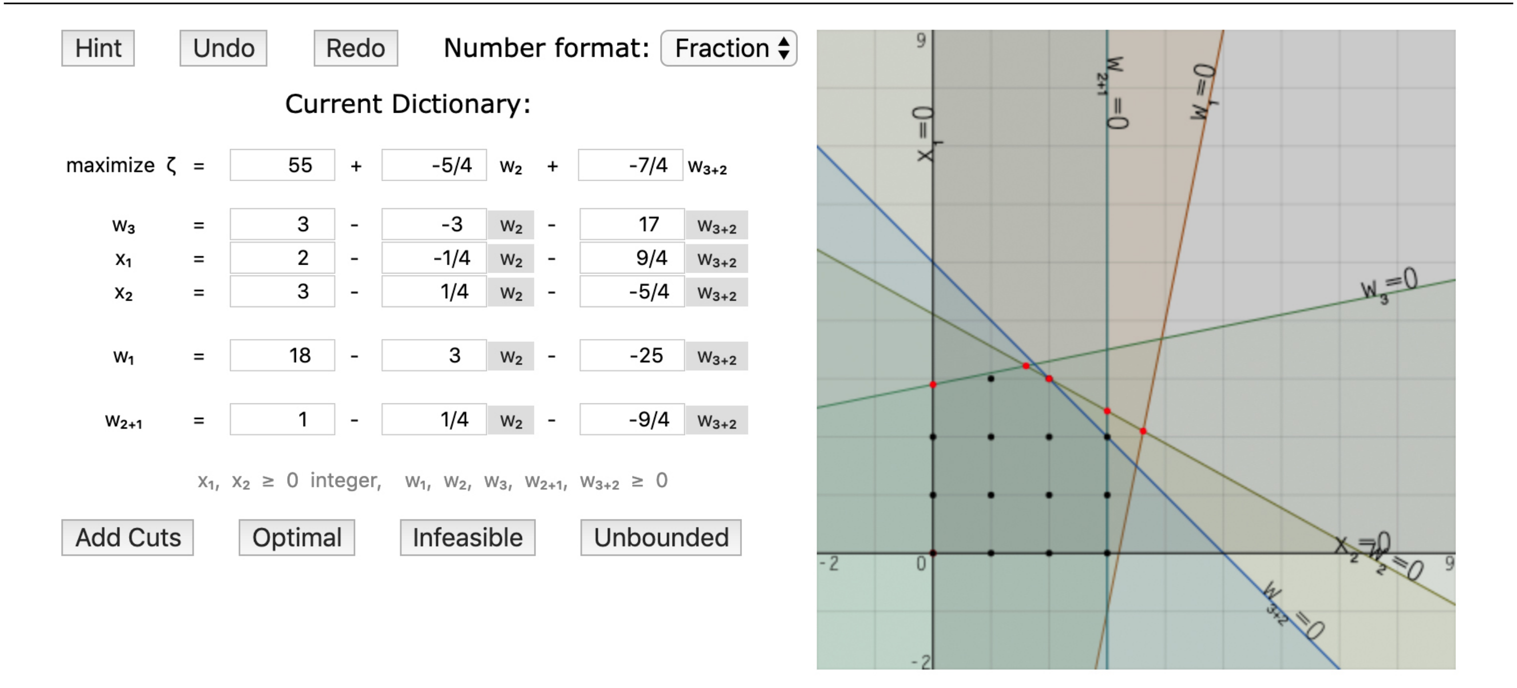
$$x_2 + 0w_2 - w_{2+1} \leq 2$$

New constraint...

$$\frac{22}{9} - \frac{1}{9}w_2 + \frac{5}{9}w_{2+1} + 0w_2 - w_{2+1} \leq 2$$



Again, we do a dual pivot:



OPTIMAL!