Satisfiability Modulo Theories
Lecture 6 - A Theory Solver for $\mathcal{LRA}$

(slides revision: Saturday 14$^{\text{th}}$ March, 2015, 11:47)

Roberto Bruttomesso

Seminario di Logica Matematica
(Corso Prof. Silvio Ghilardi)

24 Novembre 2011

Outline

1. Basic Solving
   - Introduction
   - Preprocessing
   - Solving

2. Improvement for $\mathcal{T}$-solver
   - $\mathcal{T}$-solver features
   - Strict inequalities
   - Integers

Invented by Tobias Dantzig around 1950

Used to solve optimization problems in linear programming

Greg Nelson was the first to employ it for constraint solving, AFAIK, around 1980

Difference is that in linear programming input problem is feasible and one looks for optimum. In constraint solving problem can be infeasible, and we are interested in finding any solution

Introduction

Linear Rational Arithmetic $\mathcal{LRA}$ consists in solving Boolean combinations of atoms of the form

$$\sum_{j=1}^{n} a_j x_j \leq b \qquad\qquad \sum_{j=1}^{n} a_j x_j \geq b$$

where a_j are constants (coefficients), x_j are variables, and b is a constant (bound). The domain of a_j, x_j, b is that of rationals.

Introduction

Linear Rational Arithmetic $\mathcal{LRA}$ consists in solving Boolean combinations of atoms of the form

$$\sum_{j=1}^{n} a_j x_j \leq b \qquad\qquad \sum_{j=1}^{n} a_j x_j \geq b$$

where a_j are constants (coefficients), x_j are variables, and b is a constant (bound). The domain of a_j, x_j, b is that of rationals.

Notice that the following translations hold

- $\sum_{j=1}^{n} a_j x_j = b \quad \Longrightarrow \quad (\sum_{j=1}^{n} a_j x_j \leq b) \wedge (\sum_{j=1}^{n} a_j x_j \geq b)$
- $\sum_{j=1}^{n} a_j x_j < b \quad \Longrightarrow$ see later
- $\sum_{j=1}^{n} a_j x_j > b \quad \Longrightarrow$ see later
- $\sum_{j=1}^{n} a_j x_j \neq b \quad \Longrightarrow \quad (\sum_{j=1}^{n} a_j x_j < b) \vee (\sum_{j=1}^{n} a_j x_j > b)$

Preprocessing

In the SMT setting, we are given a formula φ like

$$(x \geq 0) \wedge ((x + y \leq 2) \vee (x + 2y - z \geq 6)) \wedge ((x + y \geq 2) \vee (2y - z \leq 4))$$

We perform a **preprocessing step**, in order to separate the formula into a set of **equations** and a set of simple **bounds**. This is done by introducing **fresh** variables.

Preprocessing

In the SMT setting, we are given a formula φ like

$$(x \geq 0) \wedge ((x + y \leq 2) \vee (x + 2y - z \geq 6)) \wedge ((x + y \geq 2) \vee (2y - z \leq 4))$$

We perform a **preprocessing step**, in order to separate the formula into a set of **equations** and a set of simple **bounds**. This is done by introducing **fresh** variables.

The formula above φ is equivalent to (the conjunction of)

$$(x \geq 0) \wedge ((s_1 \leq 2) \vee (s_2 \geq 6)) \wedge ((s_1 \geq 2) \vee (s_3 \leq 4))$$

$$\begin{array}{l} (s_1 = x + y) \\ (s_2 = x + 2y - z) \\ (s_3 = 2y - z) \end{array}$$

Preprocessing

In the SMT setting, we are given a formula φ like

$$(x \geq 0) \wedge ((x + y \leq 2) \vee (x + 2y - z \geq 6)) \wedge ((x + y \geq 2) \vee (2y - z \leq 4))$$

We perform a **preprocessing step**, in order to separate the formula into a set of **equations** and a set of simple **bounds**. This is done by introducing **fresh** variables.

The formula above φ is equivalent to (the conjunction of)

$$(x \geq 0) \wedge ((s_1 \leq 2) \vee (s_2 \geq 6)) \wedge ((s_1 \geq 2) \vee (s_3 \leq 4)) \qquad \varphi'$$

$$\begin{array}{ll} (s_1 = x + y) & \\ (s_2 = x + 2y - z) & \qquad A\vec{x} = \vec{0} \\ (s_3 = 2y - z) & \end{array}$$

Preprocessing

In general, from a formula φ, we end up in a rewritten formula of the kind

$$\varphi' \ \wedge \ A\vec{x} = \vec{0}$$

where φ' is a Boolean combination of **bounds**, while $A\vec{x} = \vec{0}$ is a system of **equations** of the form

$$\begin{array}{rcl}
a_{11}x_1 + \ldots + a_{1n}x_n & = & 0 \\
a_{21}x_1 + \ldots + a_{2n}x_n & = & 0 \\
\ldots & & \\
a_{i1}x_1 + \ldots + a_{in}x_n & = & 0 \\
\ldots & & \\
a_{m1}x_1 + \ldots + a_{mn}x_n & = & 0
\end{array}$$

Preprocessing

In general, from a formula φ, we end up in a rewritten formula of the kind

$$\varphi' \ \wedge \ A\vec{x} = \vec{0}$$

where φ' is a Boolean combination of **bounds**, while $A\vec{x} = \vec{0}$ is a system of **equations** of the form

$$\begin{array}{rcl}
a_{11}x_1 + \ldots + a_{1n}x_n & = & 0 \\
a_{21}x_1 + \ldots + a_{2n}x_n & = & 0 \\
\ldots & & \\
a_{i1}x_1 + \ldots + a_{in}x_n & = & 0 \\
\ldots & & \\
a_{m1}x_1 + \ldots + a_{mn}x_n & = & 0
\end{array}$$

Now we detach $A\vec{x} = \vec{0}$ from the formula, and we store it into the $\mathcal{T}$-solver permanently. The SAT-solver will work only on φ'. Therefore **the constraints that are pushed into and popped from the $\mathcal{T}$-solver are just bounds**

The Tableau

The equations $A\vec{x} = \vec{0}$ are kept in a **tableau**, the most important structure of the Simplex

The variables are **partitioned** into the set of **non-basic** $\mathcal{N}$ and **basic** $\mathcal{B}$ variables

E.g., $\mathcal{B} = \{x_1, x_3, x_4\}$, $\mathcal{N} = \{x_2, x_5, x_6\}$

$$\begin{array}{rcl} x_1 & = & 4x_2 + x_5 \\ x_3 & = & 5x_2 + 3x_6 \\ x_4 & = & x_5 - x_6 \end{array}$$

non-basic variables can be considered as **independent**, while basic variables assume values forced by the non-basic ones. E.g., in the row

$$x_1 = 4x_2 + x_5$$

suppose that $x_2 = 2, x_5 = 1$, then we set $x_1 = 9$

Solving

The $\mathcal{T}$-solver stores

- the Tableau (does not grow/shrink)
- the active bounds on variables (initially none)
- the current model μ (initially all 0, but could be chosen differently)

$$
\begin{array}{rcl}
& \text{Tableau} & \\
x_1 & = & a_{11}x_{m+1}\ldots + a_{1n}x_n \\
x_2 & = & a_{21}x_{m+1}\ldots + a_{2n}x_n \\
& \ldots & \\
x_i & = & a_{i1}x_{m+1}\ldots + a_{in}x_n \\
& \ldots & \\
x_m & = & a_{m1}x_{m+1}\ldots + a_{mn}x_n
\end{array}
\qquad
\begin{array}{ccccc}
lb & & \text{Bounds} & & ub \\
-\infty & \leq & x_1 & \leq & \infty \\
-\infty & \leq & x_2 & \leq & \infty \\
& & \ldots & & \\
-\infty & \leq & x_i & \leq & \infty \\
& & \ldots & & \\
-\infty & \leq & x_m & \leq & \infty \\
& & \ldots & & \\
-\infty & \leq & x_n & \leq & \infty
\end{array}
\qquad
\begin{array}{rcl}
& \mu & \\
x_1 & \mapsto & 0 \\
x_2 & \mapsto & 0 \\
& \ldots & \\
x_i & \mapsto & 0 \\
& \ldots & \\
x_m & \mapsto & 0 \\
& \ldots & \\
x_n & \mapsto & 0
\end{array}
$$

The $\mathcal{T}$-solver is in a consistent state if the model (i) respects the tableau and (ii) satisfies the bounds (for all x, $lb(x) \leq \mu(x) \leq ub(x)$)

Asserting a bound on a non-basic variable

Asserting a bound $x \leq c$ (resp. $x \geq c$), $x \in \mathcal{N}$ may result in

Asserting a bound on a non-basic variable

Asserting a bound $x \leq c$ (resp. $x \geq c$), $x \in \mathcal{N}$ may result in

- unsatisfiability, if $c < lb(x)$ (resp. $c > ub(x)$)

Asserting a bound on a non-basic variable

Asserting a bound $x \leq c$ (resp. $x \geq c$), $x \in \mathcal{N}$ may result in

- unsatisfiability, if $c < lb(x)$ (resp. $c > ub(x)$)
- nothing, if $c > ub(x)$ (resp. $c < lb(x)$)

Asserting a bound on a non-basic variable

Asserting a bound $x \leq c$ (resp. $x \geq c$), $x \in \mathcal{N}$ may result in

- unsatisfiability, if $c < lb(x)$ (resp. $c > ub(x)$)
- nothing, if $c > ub(x)$ (resp. $c < lb(x)$)
- bound tightening, model not affected if $\mu(x) \leq c$ (resp. $\mu(x) \geq c$)

Asserting a bound on a non-basic variable

Asserting a bound $x \leq c$ (resp. $x \geq c$), $x \in \mathcal{N}$ may result in

- unsatisfiability, if $c < lb(x)$ (resp. $c > ub(x)$)
- nothing, if $c > ub(x)$ (resp. $c < lb(x)$)
- bound tightening, model not affected if $\mu(x) \leq c$ (resp. $\mu(x) \geq c$)
- bound tightening, model affected if $\mu(x) > c$ (resp. $\mu(x) < c$)

Asserting a bound on a non-basic variable

Asserting a bound $x \leq c$ (resp. $x \geq c$), $x \in \mathcal{N}$ may result in

- unsatisfiability, if $c < lb(x)$ (resp. $c > ub(x)$)
- nothing, if $c > ub(x)$ (resp. $c < lb(x)$)
- bound tightening, model not affected if $\mu(x) \leq c$ (resp. $\mu(x) \geq c$)
- bound tightening, model affected if $\mu(x) > c$ (resp. $\mu(x) < c$)

The last case is “problematic” because we need to

(i) adjust $\mu(x)$: $\mu(x)$ is set to c

(ii) adjust the values of basic variables

We assume we have a function $Update(x, c)$ that implements $(i) - (ii)$

Asserting a bound on a non-basic variable

Asserting a bound $x \leq c$ (resp. $x \geq c$), $x \in \mathcal{N}$ may result in

- unsatisfiability, if $c < lb(x)$ (resp. $c > ub(x)$)
- nothing, if $c > ub(x)$ (resp. $c < lb(x)$)
- bound tightening, model not affected if $\mu(x) \leq c$ (resp. $\mu(x) \geq c$)
- bound tightening, model affected if $\mu(x) > c$ (resp. $\mu(x) < c$)

The last case is "problematic" because we need to

(*i*) adjust $\mu(x)$: $\mu(x)$ is set to c

(*ii*) adjust the values of basic variables

We assume we have a function $Update(x, c)$ that implements $(i) - (ii)$

Tableau	lb	Bounds	ub	μ
$x_1 = -x_3 + x_4$	$-\infty \leq$	x_1	$\leq \infty$	$x_1 \mapsto 0$
$x_2 = x_3 + x_4$	$-\infty \leq$	x_2	$\leq \infty$	$x_2 \mapsto 0$
	$-\infty \leq$	x_3	$\leq \infty$	$x_3 \mapsto 0$
	$-\infty \leq$	x_4	$\leq \infty$	$x_4 \mapsto 0$

$\mathcal{T}$-solver stack:

Asserting a bound on a non-basic variable

Asserting a bound $x \leq c$ (resp. $x \geq c$), $x \in \mathcal{N}$ may result in

- unsatisfiability, if $c < lb(x)$ (resp. $c > ub(x)$)
- nothing, if $c > ub(x)$ (resp. $c < lb(x)$)
- bound tightening, model not affected if $\mu(x) \leq c$ (resp. $\mu(x) \geq c$)
- bound tightening, model affected if $\mu(x) > c$ (resp. $\mu(x) < c$)

The last case is "problematic" because we need to

(i) adjust $\mu(x)$: $\mu(x)$ is set to c

(ii) adjust the values of basic variables

We assume we have a function $Update(x, c)$ that implements $(i) - (ii)$

Tableau	lb	Bounds	ub	μ
$x_1 = -x_3 + x_4$	$-\infty \leq$	x_1	$\leq \infty$	$x_1 \mapsto 0$
$x_2 = x_3 + x_4$	$-\infty \leq$	x_2	$\leq \infty$	$x_2 \mapsto 0$
	$-\infty \leq$	x_3	≤ -4	$x_3 \mapsto 0$
	$-\infty \leq$	x_4	$\leq \infty$	$x_4 \mapsto 0$

$\mathcal{T}$-solver stack:

$x_3 \leq -4$ (tighten $ub(x_3)$, affects other values)

Asserting a bound on a non-basic variable

Asserting a bound $x \leq c$ (resp. $x \geq c$), $x \in \mathcal{N}$ may result in

- unsatisfiability, if $c < lb(x)$ (resp. $c > ub(x)$)
- nothing, if $c > ub(x)$ (resp. $c < lb(x)$)
- bound tightening, model not affected if $\mu(x) \leq c$ (resp. $\mu(x) \geq c$)
- bound tightening, model affected if $\mu(x) > c$ (resp. $\mu(x) < c$)

The last case is "problematic" because we need to

(i) adjust $\mu(x)$: $\mu(x)$ is set to c

(ii) adjust the values of basic variables

We assume we have a function $Update(x, c)$ that implements $(i) - (ii)$

Tableau

$$\begin{array}{rcl} x_1 & = & -x_3 + x_4 \\ x_2 & = & x_3 + x_4 \end{array}$$

lb		Bounds		ub
$-\infty$	$\leq$	x_1	$\leq$	∞
$-\infty$	$\leq$	x_2	$\leq$	∞
$-\infty$	$\leq$	x_3	$\leq$	-4
$-\infty$	$\leq$	x_4	$\leq$	∞

μ

$$\begin{array}{rcl} x_1 & \mapsto & 4 \\ x_2 & \mapsto & -4 \\ x_3 & \mapsto & -4 \\ x_4 & \mapsto & 0 \end{array}$$

$\mathcal{T}$-solver stack:

$x_3 \leq -4$ (tighten $ub(x_3)$, affects other values)

Asserting a bound on a non-basic variable

Asserting a bound $x \leq c$ (resp. $x \geq c$), $x \in \mathcal{N}$ may result in

- unsatisfiability, if $c < lb(x)$ (resp. $c > ub(x)$)
- nothing, if $c > ub(x)$ (resp. $c < lb(x)$)
- bound tightening, model not affected if $\mu(x) \leq c$ (resp. $\mu(x) \geq c$)
- bound tightening, model affected if $\mu(x) > c$ (resp. $\mu(x) < c$)

The last case is "problematic" because we need to

(i) adjust $\mu(x)$: $\mu(x)$ is set to c

(ii) adjust the values of basic variables

We assume we have a function $Update(x, c)$ that implements $(i) - (ii)$

Tableau			lb	Bounds			ub	μ		
x_1	$=$	$-x_3 + x_4$	$-\infty$	$\leq$	x_1	$\leq$	∞	x_1	$\mapsto$	4
x_2	$=$	$x_3 + x_4$	$-\infty$	$\leq$	x_2	$\leq$	∞	x_2	$\mapsto$	-4
			-8	$\leq$	x_3	$\leq$	-4	x_3	$\mapsto$	-4
			$-\infty$	$\leq$	x_4	$\leq$	∞	x_4	$\mapsto$	0

$\mathcal{T}$-solver stack:

$x_3 \leq -4$ (tighten $ub(x_3)$, affects other values)

$x_3 \geq -8$ (tighten $lb(x_3)$, does not affect other values)

Asserting a bound on a non-basic variable

Asserting a bound $x \leq c$ (resp. $x \geq c$), $x \in \mathcal{N}$ may result in

- unsatisfiability, if $c < lb(x)$ (resp. $c > ub(x)$)
- nothing, if $c > ub(x)$ (resp. $c < lb(x)$)
- bound tightening, model not affected if $\mu(x) \leq c$ (resp. $\mu(x) \geq c$)
- bound tightening, model affected if $\mu(x) > c$ (resp. $\mu(x) < c$)

The last case is "problematic" because we need to

(i) adjust $\mu(x)$: $\mu(x)$ is set to c

(ii) adjust the values of basic variables

We assume we have a function $Update(x, c)$ that implements $(i) - (ii)$

Tableau			lb		Bounds		ub		μ	
x_1	$=$	$-x_3 + x_4$	$-\infty$	$\leq$	x_1	$\leq$	∞	x_1	$\mapsto$	4
x_2	$=$	$x_3 + x_4$	$-\infty$	$\leq$	x_2	$\leq$	∞	x_2	$\mapsto$	-4
			-8	$\leq$	x_3	$\leq$	-4	x_3	$\mapsto$	-4
			$-\infty$	$\leq$	x_4	$\leq$	∞	x_4	$\mapsto$	0

$\mathcal{T}$-solver stack:

$x_3 \leq -4$ (tighten $ub(x_3)$, affects other values)

$x_3 \geq -8$ (tighten $lb(x_3)$, does not affect other values)

$x_3 \leq 0$ (does not tighten $ub(x_3)$)

Asserting a bound on a basic variable

Asserting a bound $x \leq c$ (resp. $x \geq c$), $x \in \mathcal{B}$ may result in the same 4 cases as before

- unsatisfiability, if $c < lb(x)$ (resp. $c > ub(x)$)
- nothing, if $c > ub(x)$ (resp. $c < lb(x)$)
- bound tightening, model not affected if $\mu(x) \leq c$ (resp. $\mu(x) \geq c$)
- bound tightening, model affected if $\mu(x) > c$ (resp. $\mu(x) < c$)

however, since a basic variable is **dependent** from non-basic variables in the tableau, we cannot use function *Update* directly. Before we need to turn x into a non-basic variables.

Asserting a bound on a basic variable

Asserting a bound $x \leq c$ (resp. $x \geq c$), $x \in \mathcal{B}$ may result in the same 4 cases as before

- unsatisfiability, if $c < lb(x)$ (resp. $c > ub(x)$)
- nothing, if $c > ub(x)$ (resp. $c < lb(x)$)
- bound tightening, model not affected if $\mu(x) \leq c$ (resp. $\mu(x) \geq c$)
- bound tightening, model affected if $\mu(x) > c$ (resp. $\mu(x) < c$)

however, since a basic variable is **dependent** from non-basic variables in the tableau, we cannot use function *Update* directly. Before we need to turn x into a non-basic variables.

So if $\mu(x) > c$ or $\mu(x) < c$ we have to

(i) **turn x into non-basic** (another non-basic variable will become basic instead)
(ii) adjust $\mu(x)$: $\mu(x)$ is set to c
(iii) adjust the values of basic variables

Step (i) is performed by a function $Pivot(x, y)$ (see next slide). Therefore asserting a bound on a basic variable x consists in executing $Pivot(x, y)$ and then $Update(x, c)$

Pivoting

$Pivot(x, y)$ is the operation of swapping a basic variable x with a non-basic variable y (how to choose y ? See next slide)

It consists of the following steps:

1. Take the row $x = ay + R$ in the tableau ($R =$ rest of the polynome)
2. Rewrite it as $y = \frac{x-R}{a}$
3. Substitute y with $\frac{x-R}{a}$ in all the other rows of the tableau (and simplify)

Pivoting

$Pivot(x, y)$ is the operation of swapping a basic variable x with a non-basic variable y (how to choose y ? See next slide)

It consists of the following steps:

1. Take the row $x = ay + R$ in the tableau ($R =$ rest of the polynome)
2. Rewrite it as $y = \frac{x-R}{a}$
3. Substitute y with $\frac{x-R}{a}$ in all the other rows of the tableau (and simplify)

Example for $Pivot(x_1, x_3)$

Step 1	Step 2	Step 3
$x_1 = 3x_2 + 4x_3 - 5x_4$	$x_3 = -\frac{3}{4}x_2 + \frac{1}{4}x_1 + \frac{5}{4}x_4$	$x_3 = -\frac{3}{4}x_2 + \frac{1}{4}x_1 + \frac{5}{4}x_4$
$x_5 = -x_2 - x_3$	$x_5 = -x_2 - x_3$	$x_5 = -\frac{1}{4}x_2 - \frac{1}{4}x_1 - \frac{5}{4}x_4$
$x_6 = 10x_2 + 5x_4$	$x_6 = 10x_2 + 5x_4$	$x_6 = 10x_2 + 5x_4$

Pivoting

$Pivot(x, y)$ is the operation of swapping a basic variable x with a non-basic variable y (how to choose y ? See next slide)

It consists of the following steps:

1. Take the row $x = ay + R$ in the tableau ($R =$ rest of the polynome)
2. Rewrite it as $y = \frac{x-R}{a}$
3. Substitute y with $\frac{x-R}{a}$ in all the other rows of the tableau (and simplify)

Example for $Pivot(x_1, x_3)$

Step 1	Step 2	Step 3
$x_1 = 3x_2 + 4x_3 - 5x_4$	$x_3 = -\frac{3}{4}x_2 + \frac{1}{4}x_1 + \frac{5}{4}x_4$	$x_3 = -\frac{3}{4}x_2 + \frac{1}{4}x_1 + \frac{5}{4}x_4$
$x_5 = -x_2 - x_3$	$x_5 = -x_2 - x_3$	$x_5 = -\frac{1}{4}x_2 - \frac{1}{4}x_1 - \frac{5}{4}x_4$
$x_6 = 10x_2 + 5x_4$	$x_6 = 10x_2 + 5x_4$	$x_6 = 10x_2 + 5x_4$

we moved from $\mathcal{B} = \{x_1, x_5, x_6\}$ to $\mathcal{B} = \{x_3, x_5, x_6\}$

Choosing Pivoting Variable

Consider the following situation

Tableau			*lb*		Bounds		*ub*		μ	
	$\dots$		-4	$\leq$	x_1	$\leq$	10	x_1	$\mapsto$	12
x_1	$=$	$3x_2 - 4x_3 + 2x_4 - x_5$	1	$\leq$	x_2	$\leq$	3	x_2	$\mapsto$	1
	$\dots$		-4	$\leq$	x_3	$\leq$	-1	x_3	$\mapsto$	-1
			1	$\leq$	x_4	$\leq$	2	x_4	$\mapsto$	2
			-1	$\leq$	x_5	$\leq$	10	x_5	$\mapsto$	-1

which among $\mathcal{N} = \{x_2, x_3, x_4\}$ do I choose for pivoting ? Clearly, the value of $\mu(x_1)$ is too high, I have to decrease it by playing with the values of $\mathcal{N}$:

Choosing Pivoting Variable

Consider the following situation

Tableau			lb		Bounds		ub		μ	
	$\dots$		-4	$\leq$	x_1	$\leq$	10	x_1	$\mapsto$	12
x_1	$=$	$3x_2 - 4x_3 + 2x_4 - x_5$	1	$\leq$	x_2	$\leq$	3	x_2	$\mapsto$	1
	$\dots$		-4	$\leq$	x_3	$\leq$	-1	x_3	$\mapsto$	-1
			1	$\leq$	x_4	$\leq$	2	x_4	$\mapsto$	2
			-1	$\leq$	x_5	$\leq$	10	x_5	$\mapsto$	-1

which among $\mathcal{N} = \{x_2, x_3, x_4\}$ do I choose for pivoting ? Clearly, the value of $\mu(x_1)$ is too high, I have to decrease it by playing with the values of $\mathcal{N}$:

- $3x_2$ cannot decrease, as $\mu(x_2) = lb(x_2)$ and cannot be moved down

Choosing Pivoting Variable

Consider the following situation

Tableau			lb		Bounds		ub		μ	
	$\dots$									
x_1	$=$	$3x_2 - 4x_3 + 2x_4 - x_5$	-4	$\leq$	x_1	$\leq$	10	x_1	$\mapsto$	12
	$\dots$		1	$\leq$	x_2	$\leq$	3	x_2	$\mapsto$	1
			-4	$\leq$	x_3	$\leq$	-1	x_3	$\mapsto$	-1
			1	$\leq$	x_4	$\leq$	2	x_4	$\mapsto$	2
			-1	$\leq$	x_5	$\leq$	10	x_5	$\mapsto$	-1

which among $\mathcal{N} = \{x_2, x_3, x_4\}$ do I choose for pivoting ? Clearly, the value of $\mu(x_1)$ is too high, I have to decrease it by playing with the values of $\mathcal{N}$:

- $3x_2$ cannot decrease, as $\mu(x_2) = lb(x_2)$ and cannot be moved down
- $-4x_3$ cannot decrease, as $\mu(x_3) = ub(x_3)$ and cannot be moved up

Choosing Pivoting Variable

Consider the following situation

Tableau			*lb*		Bounds		*ub*		μ	
	$\ldots$		-4	$\leq$	x_1	$\leq$	10	x_1	$\mapsto$	12
x_1	$=$	$3x_2 - 4x_3 + 2x_4 - x_5$	1	$\leq$	x_2	$\leq$	3	x_2	$\mapsto$	1
	$\ldots$		-4	$\leq$	x_3	$\leq$	-1	x_3	$\mapsto$	-1
			1	$\leq$	x_4	$\leq$	2	x_4	$\mapsto$	2
			-1	$\leq$	x_5	$\leq$	10	x_5	$\mapsto$	-1

which among $\mathcal{N} = \{x_2, x_3, x_4\}$ do I choose for pivoting ? Clearly, the value of $\mu(x_1)$ is too high, I have to decrease it by playing with the values of $\mathcal{N}$:

- $3x_2$ cannot decrease, as $\mu(x_2) = lb(x_2)$ and cannot be moved down
- $-4x_3$ cannot decrease, as $\mu(x_3) = ub(x_3)$ and cannot be moved up
- $2x_4$ can decrease, as $\mu(x_4) = ub(x_4)$, and can be moved down

Choosing Pivoting Variable

Consider the following situation

Tableau			*lb*		Bounds		*ub*	μ		
	$\dots$		-4	$\leq$	x_1	$\leq$	10	x_1	$\mapsto$	12
x_1	$=$	$3x_2 - 4x_3 + 2x_4 - x_5$	1	$\leq$	x_2	$\leq$	3	x_2	$\mapsto$	1
	$\dots$		-4	$\leq$	x_3	$\leq$	-1	x_3	$\mapsto$	-1
			1	$\leq$	x_4	$\leq$	2	x_4	$\mapsto$	2
			-1	$\leq$	x_5	$\leq$	10	x_5	$\mapsto$	-1

which among $\mathcal{N} = \{x_2, x_3, x_4\}$ do I choose for pivoting ? Clearly, the value of $\mu(x_1)$ is too high, I have to decrease it by playing with the values of $\mathcal{N}$:

- $3x_2$ cannot decrease, as $\mu(x_2) = lb(x_2)$ and cannot be moved down
- $-4x_3$ cannot decrease, as $\mu(x_3) = ub(x_3)$ and cannot be moved up
- $2x_4$ can decrease, as $\mu(x_4) = ub(x_4)$, and can be moved down
- $-x_5$ can decrease, as $\mu(x_5) = lb(x_5)$, and can be moved up

Choosing Pivoting Variable

Consider the following situation

Tableau			*lb*		Bounds		*ub*		μ	
	$\dots$		-4	$\leq$	x_1	$\leq$	10	x_1	$\mapsto$	12
x_1	$=$	$3x_2 - 4x_3 + 2x_4 - x_5$	1	$\leq$	x_2	$\leq$	3	x_2	$\mapsto$	1
	$\dots$		-4	$\leq$	x_3	$\leq$	-1	x_3	$\mapsto$	-1
			1	$\leq$	x_4	$\leq$	2	x_4	$\mapsto$	2
			-1	$\leq$	x_5	$\leq$	10	x_5	$\mapsto$	-1

which among $\mathcal{N} = \{x_2, x_3, x_4\}$ do I choose for pivoting ? Clearly, the value of $\mu(x_1)$ is too high, I have to decrease it by playing with the values of $\mathcal{N}$:

- $3x_2$ cannot decrease, as $\mu(x_2) = lb(x_2)$ and cannot be moved down
- $-4x_3$ cannot decrease, as $\mu(x_3) = ub(x_3)$ and cannot be moved up
- $2x_4$ can decrease, as $\mu(x_4) = ub(x_4)$, and can be moved down
- $-x_5$ can decrease, as $\mu(x_5) = lb(x_5)$, and can be moved up

both x_4 and x_5 are therefore good candidates for pivoting. To avoid loops, choose variable with smallest subscript (Bland's Rule). This rule is not necessarily efficient, though

Detect Unsatisfiability

There might be cases in which **no suitable variable for pivoting can be found**. This indicates unsatisfiability.

Detect Unsatisfiability

There might be cases in which **no suitable variable for pivoting can be found**. This indicates unsatisfiability. Consider the following where we have just asserted $x_1 \leq 9$

Tableau			*lb*		Bounds		*ub*		μ	
	$\dots$		-4	$\leq$	x_1	$\leq$	9	x_1	$\mapsto$	12
x_1	$=$	$3x_2 - 4x_3 + 2x_4 - x_5$	1	$\leq$	x_2	$\leq$	3	x_2	$\mapsto$	1
	$\dots$		-4	$\leq$	x_3	$\leq$	-1	x_3	$\mapsto$	-1
			2	$\leq$	x_4	$\leq$	2	x_4	$\mapsto$	2
			-1	$\leq$	x_5	$\leq$	-1	x_5	$\mapsto$	-1

no variable among $\mathcal{N} = \{x_2, x_3, x_4\}$ can be chosen for pivoting. This is because (due to tableau)

$$x_2 \geq 3 \;\wedge\; x_3 \leq -1 \;\wedge\; x_4 \geq 4 \;\wedge\; x_5 \leq -1 \;\Rightarrow\; x_1 \geq 12 \;\Rightarrow\; \neg(x_1 \leq 9)$$

Therefore

$$\{x_2 \geq 3,\ x_3 \leq -1,\ x_4 \geq 4,\ x_5 \leq -1,\ \neg(x_1 \leq 9)\}$$

is a $\mathcal{T}$-conflict (modulo the tableau)

Solving

```
while( true )

  pick first x_i ∈ B such that μ(x_i) < lb(x_i) or μ(x_i) > ub(x_i)
  if ( there is no such x_i ) return sat

  x_j = ChoosePivot(x_i)
  if ( x_j == undef ) return unsat
  Pivot(x_i, x_j)

  if ( μ(x_i) < lb(x_i) )
    Update(x_i, lb(x_i))

  if ( μ(x_i) > ub(x_i) )
    Update(x_i, ub(x_i))

end
```

$x_j = ChoosePivot(x_i)$: returns variable to use for pivoting with x_i, or $undef$ if conflict
pick first x_i ...: it's again Bland's rule

An Example

Consider the following problem

$$x_1 - x_2 \leq 8 \ \wedge \ x_2 - x_3 \leq -1 \ \wedge \ x_3 - x_4 \leq 2 \ \wedge \ x_4 - x_1 \leq -10$$

An Example

Consider the following problem

$$x_5 \leq 8 \;\wedge\; x_6 \leq -1 \;\wedge\; x_7 \leq 2 \;\wedge\; x_8 \leq -10$$

Tableau

$$\begin{array}{rcl} x_5 & = & x_1 - x_2 \\ x_6 & = & x_2 - x_3 \\ x_7 & = & x_3 - x_4 \\ x_8 & = & x_4 - x_1 \end{array}$$

lb Bounds ub

$$\begin{array}{ccccc} -\infty & \leq & x_5 & \leq & \infty \\ -\infty & \leq & x_6 & \leq & \infty \\ -\infty & \leq & x_7 & \leq & \infty \\ -\infty & \leq & x_8 & \leq & \infty \end{array}$$

μ

$$\begin{array}{ccc} x_1 & \mapsto & 0 \\ x_2 & \mapsto & 0 \\ x_3 & \mapsto & 0 \\ x_4 & \mapsto & 0 \\ x_5 & \mapsto & 0 \\ x_6 & \mapsto & 0 \\ x_7 & \mapsto & 0 \\ x_8 & \mapsto & 0 \end{array}$$

An Example

Consider the following problem

$$x_5 \leq 8 \;\wedge\; x_6 \leq -1 \;\wedge\; x_7 \leq 2 \;\wedge\; x_8 \leq -10$$

Tableau

$$\begin{array}{rcl} x_5 & = & x_1 - x_2 \\ x_6 & = & x_2 - x_3 \\ x_7 & = & x_3 - x_4 \\ x_8 & = & x_4 - x_1 \end{array}$$

Bounds

$$\begin{array}{ccccc} lb & & & & ub \\ -\infty & \leq & x_5 & \leq & 8 \\ -\infty & \leq & x_6 & \leq & \infty \\ -\infty & \leq & x_7 & \leq & \infty \\ -\infty & \leq & x_8 & \leq & \infty \end{array}$$

μ

$$\begin{array}{ccc} x_1 & \mapsto & 0 \\ x_2 & \mapsto & 0 \\ x_3 & \mapsto & 0 \\ x_4 & \mapsto & 0 \\ x_5 & \mapsto & 0 \\ x_6 & \mapsto & 0 \\ x_7 & \mapsto & 0 \\ x_8 & \mapsto & 0 \end{array}$$

Receiving $x_5 \leq 8$: OK

An Example

Consider the following problem

$$x_5 \leq 8 \ \wedge \ x_6 \leq -1 \ \wedge \ x_7 \leq 2 \ \wedge \ x_8 \leq -10$$

Tableau

$$\begin{array}{lcl} x_5 & = & x_1 - x_2 \\ x_6 & = & x_2 - x_3 \\ x_7 & = & x_3 - x_4 \\ x_8 & = & x_4 - x_1 \end{array}$$

lb Bounds ub

$$\begin{array}{ccccc} -\infty & \leq & x_5 & \leq & 8 \\ -\infty & \leq & x_6 & \leq & -1 \\ -\infty & \leq & x_7 & \leq & \infty \\ -\infty & \leq & x_8 & \leq & \infty \end{array}$$

μ

$$\begin{array}{lcl} x_1 & \mapsto & 0 \\ x_2 & \mapsto & 0 \\ x_3 & \mapsto & 0 \\ x_4 & \mapsto & 0 \\ x_5 & \mapsto & 0 \\ x_6 & \mapsto & 0 \\ x_7 & \mapsto & 0 \\ x_8 & \mapsto & 0 \end{array}$$

Receiving $x_6 \leq -1$: $\mu(x_6)$ out of bounds

An Example

Consider the following problem

$$x_5 \leq 8 \;\wedge\; x_6 \leq -1 \;\wedge\; x_7 \leq 2 \;\wedge\; x_8 \leq -10$$

$$\begin{array}{ccc}
 & \text{Tableau} & \\
 & & \\
x_5 & = & x_1 - x_2 \\
x_3 & = & x_2 - x_6 \\
x_7 & = & x_2 - x_6 - x_4 \\
x_8 & = & x_4 - x_1
\end{array}
\qquad
\begin{array}{ccccc}
lb & & \text{Bounds} & & ub \\
 & & & & \\
-\infty & \leq & x_5 & \leq & 8 \\
-\infty & \leq & x_6 & \leq & -1 \\
-\infty & \leq & x_7 & \leq & \infty \\
-\infty & \leq & x_8 & \leq & \infty
\end{array}
\qquad
\begin{array}{ccc}
 & \mu & \\
x_1 & \mapsto & 0 \\
x_2 & \mapsto & 0 \\
x_3 & \mapsto & 1 \\
x_4 & \mapsto & 0 \\
x_5 & \mapsto & 0 \\
x_6 & \mapsto & -1 \\
x_7 & \mapsto & 1 \\
x_8 & \mapsto & 0
\end{array}$$

$Pivot(x_6, x_3)$, $Update(x_6, -1)$. Values of x_3 and x_7 change as well

An Example

Consider the following problem

$$x_5 \leq 8 \;\wedge\; x_6 \leq -1 \;\wedge\; x_7 \leq 2 \;\wedge\; x_8 \leq -10$$

Tableau

$$\begin{array}{lcl} x_5 & = & x_1 - x_2 \\ x_3 & = & x_2 - x_6 \\ x_7 & = & x_2 - x_6 - x_4 \\ x_8 & = & x_4 - x_1 \end{array}$$

Bounds

$$\begin{array}{ccccc} lb & & & & ub \\ -\infty & \leq & x_5 & \leq & 8 \\ -\infty & \leq & x_6 & \leq & -1 \\ -\infty & \leq & x_7 & \leq & 2 \\ -\infty & \leq & x_8 & \leq & \infty \end{array}$$

μ

$$\begin{array}{lcl} x_1 & \mapsto & 0 \\ x_2 & \mapsto & 0 \\ x_3 & \mapsto & 1 \\ x_4 & \mapsto & 0 \\ x_5 & \mapsto & 0 \\ x_6 & \mapsto & -1 \\ x_7 & \mapsto & 1 \\ x_8 & \mapsto & 0 \end{array}$$

Receiving $x_7 \leq 2$: OK

An Example

Consider the following problem

$$x_5 \leq 8 \;\wedge\; x_6 \leq -1 \;\wedge\; x_7 \leq 2 \;\wedge\; x_8 \leq -10$$

Tableau

$$\begin{array}{lcl} x_5 & = & x_1 - x_2 \\ x_3 & = & x_2 - x_6 \\ x_7 & = & x_2 - x_6 - x_4 \\ x_8 & = & x_4 - x_1 \end{array}$$

lb Bounds ub

$$\begin{array}{ccccc} -\infty & \leq & x_5 & \leq & 8 \\ -\infty & \leq & x_6 & \leq & -1 \\ -\infty & \leq & x_7 & \leq & 2 \\ -\infty & \leq & x_8 & \leq & -10 \end{array}$$

μ

$$\begin{array}{lcl} x_1 & \mapsto & 0 \\ x_2 & \mapsto & 0 \\ x_3 & \mapsto & 1 \\ x_4 & \mapsto & 0 \\ x_5 & \mapsto & 0 \\ x_6 & \mapsto & -1 \\ x_7 & \mapsto & 1 \\ x_8 & \mapsto & 0 \end{array}$$

Receiving $x_8 \leq -10$: $\mu(x_8)$ out of bounds

An Example

Consider the following problem

$$x_5 \leq 8 \;\wedge\; x_6 \leq -1 \;\wedge\; x_7 \leq 2 \;\wedge\; x_8 \leq -10$$

$$\begin{array}{ccc}
 & \text{Tableau} & \\
x_5 & = & x_4 - x_8 - x_2 \\
x_3 & = & x_2 - x_6 \\
x_7 & = & x_2 - x_6 - x_4 \\
x_1 & = & x_4 - x_8
\end{array}
\qquad
\begin{array}{ccccc}
lb & & \text{Bounds} & & ub \\
-\infty & \leq & x_5 & \leq & 8 \\
-\infty & \leq & x_6 & \leq & -1 \\
-\infty & \leq & x_7 & \leq & 2 \\
-\infty & \leq & x_8 & \leq & -10
\end{array}
\qquad
\begin{array}{ccc}
 & \mu & \\
x_1 & \mapsto & 10 \\
x_2 & \mapsto & 0 \\
x_3 & \mapsto & 1 \\
x_4 & \mapsto & 0 \\
x_5 & \mapsto & 10 \\
x_6 & \mapsto & -1 \\
x_7 & \mapsto & 1 \\
x_8 & \mapsto & -10
\end{array}$$

$Pivot(x_8, x_1)$, $Update(x_8, -10)$. Values of x_5 and x_1 change as well. $\mu(x_5)$ out of bounds

An Example

Consider the following problem

$$x_5 \leq 8 \;\wedge\; x_6 \leq -1 \;\wedge\; x_7 \leq 2 \;\wedge\; x_8 \leq -10$$

Tableau

$$\begin{array}{rcl} x_2 & = & x_4 - x_8 - x_5 \\ x_3 & = & x_4 - x_8 - x_5 - x_6 \\ x_7 & = & -x_5 - x_8 - x_6 \\ x_1 & = & x_4 - x_8 \end{array}$$

$$\begin{array}{ccccc} lb & & \text{Bounds} & & ub \\ -\infty & \leq & x_5 & \leq & 8 \\ -\infty & \leq & x_6 & \leq & -1 \\ -\infty & \leq & x_7 & \leq & 2 \\ -\infty & \leq & x_8 & \leq & -10 \end{array}$$

μ

$$\begin{array}{lcl} x_1 & \mapsto & 10 \\ x_2 & \mapsto & 2 \\ x_3 & \mapsto & 1 \\ x_4 & \mapsto & 0 \\ x_5 & \mapsto & 8 \\ x_6 & \mapsto & -1 \\ x_7 & \mapsto & 3 \\ x_8 & \mapsto & -10 \end{array}$$

$Pivot(x_5, x_2)$, $Update(x_5, 8)$. Values of x_2 and x_7 change as well. $\mu(x_7)$ out of bounds

An Example

Consider the following problem

$$x_5 \leq 8 \;\wedge\; x_6 \leq -1 \;\wedge\; x_7 \leq 2 \;\wedge\; x_8 \leq -10$$

$$\begin{array}{lcl}
 & \text{Tableau} & \\
x_2 & = & x_4 - x_8 - x_5 \\
x_3 & = & x_4 - x_8 - x_5 - x_6 \\
\color{red}{x_7} & \color{red}{=} & \color{red}{-x_5 - x_8 - x_6} \\
x_1 & = & x_4 - x_8
\end{array}
\qquad
\begin{array}{ccccc}
lb & & \text{Bounds} & & ub \\
-\infty & \leq & x_5 & \leq & 8 \\
-\infty & \leq & x_6 & \leq & -1 \\
-\infty & \leq & x_7 & \leq & 2 \\
-\infty & \leq & x_8 & \leq & -10
\end{array}
\qquad
\begin{array}{lcl}
 & \mu & \\
x_1 & \mapsto & 10 \\
x_2 & \mapsto & 2 \\
x_3 & \mapsto & 1 \\
x_4 & \mapsto & 0 \\
x_5 & \mapsto & 8 \\
x_6 & \mapsto & -1 \\
x_7 & \mapsto & 3 \\
x_8 & \mapsto & -10
\end{array}$$

Cannot find suitable variable for pivoting ($ChoosePivot(x_7) == undef$). Return *unsat*

Outline

1. Basic Solving
   - Introduction
   - Preprocessing
   - Solving

2. Improvement for $\mathcal{T}$-solver
   - $\mathcal{T}$-solver features
   - Strict inequalities
   - Integers

$\mathcal{T}$-solver features

- Incrementality: it comes for free, as we keep μ updated

$\mathcal{T}$-solver features

- Incrementality: it comes for free, as we keep μ updated
- Backtrackability: use the same trick as for BF: update a model μ', if satisfiable set $\mu = \mu'$, otherwise forget μ'. When backtracking don't change μ

$\mathcal{T}$-solver features

- Incrementality: it comes for free, as we keep μ updated
- Backtrackability: use the same trick as for BF: update a model μ', if satisfiable set $\mu = \mu'$, otherwise forget μ'. When backtracking don't change μ
- Minimal $\mathcal{T}$-conflicts: seen already

$\mathcal{T}$-solver features

- Incrementality: it comes for free, as we keep μ updated
- Backtrackability: use the same trick as for BF: update a model μ', if satisfiable set $\mu = \mu'$, otherwise forget μ'. When backtracking don't change μ
- Minimal $\mathcal{T}$-conflicts: seen already
- Theory Propagation:
  - Bound propagation (cheap): if $x \leq c$ has been asserted, then all other inactive $x \leq c'$ with $c \leq c'$ are implied (similar for $\geq$)

$\mathcal{T}$-solver features

- Incrementality: it comes for free, as we keep μ updated
- Backtrackability: use the same trick as for BF: update a model μ', if satisfiable set $\mu = \mu'$, otherwise forget μ'. When backtracking don't change μ
- Minimal $\mathcal{T}$-conflicts: seen already
- Theory Propagation:
    - Bound propagation (cheap): if $x \le c$ has been asserted, then all other inactive $x \le c'$ with $c \le c'$ are implied (similar for $\ge$)
    - Tableau+Bound propagation: use a row $x_1 = a_2 x_2 + a_3 x_3 + \ldots$ and bounds on $x_2, x_3, \ldots$ to derive bounds on x_1 (similar idea as to find conflicts)

Strict inequalities

So far we have used bounds like $x \leq 3$, but never strict ones like $x < 3$

Strict inequalities

So far we have used bounds like $x \leq 3$, but never strict ones like $x < 3$

Since we are on the rationals $x < 3$ is same as $x \leq (3 - \delta)$, where δ is a symbolic positive parameter $(\delta > 0)$

Strict inequalities

So far we have used bounds like $x \leq 3$, but never strict ones like $x < 3$

Since we are on the rationals $x < 3$ is same as $x \leq (3 - \delta)$, where δ is a symbolic positive parameter ($\delta > 0$)

We do the following: instead of using $\mathbb{Q}$ numbers of form n, we use $\mathbb{Q}_\delta$ numbers of form (n, k) to be intended as $n + k\delta$

Strict inequalities

So far we have used bounds like $x \leq 3$, but never strict ones like $x < 3$

Since we are on the rationals $x < 3$ is same as $x \leq (3 - \delta)$, where δ is a symbolic positive parameter ($\delta > 0$)

We do the following: instead of using $\mathbb{Q}$ numbers of form n, we use $\mathbb{Q}_\delta$ numbers of form (n, k) to be intended as $n + k\delta$

Addition in $\mathbb{Q}_\delta$: $(n_1, k_1) + (n_2, k_2) = (n_1 + n_2, k_1 + k_2)$

Strict inequalities

So far we have used bounds like $x \leq 3$, but never strict ones like $x < 3$

Since we are on the rationals $x < 3$ is same as $x \leq (3 - \delta)$, where δ is a symbolic positive parameter ($\delta > 0$)

We do the following: instead of using $\mathbb{Q}$ numbers of form n, we use $\mathbb{Q}_\delta$ numbers of form (n, k) to be intended as $n + k\delta$

Addition in $\mathbb{Q}_\delta$: $(n_1, k_1) + (n_2, k_2) = (n_1 + n_2, k_1 + k_2)$

Scalar multiplication in $\mathbb{Q}_\delta$: $c(n, k) = (cn, ck)$

Strict inequalities

So far we have used bounds like $x \leq 3$, but never strict ones like $x < 3$

Since we are on the rationals $x < 3$ is same as $x \leq (3 - \delta)$, where δ is a symbolic positive parameter ($\delta > 0$)

We do the following: instead of using $\mathbb{Q}$ numbers of form n, we use $\mathbb{Q}_\delta$ numbers of form (n, k) to be intended as $n + k\delta$

Addition in $\mathbb{Q}_\delta$: $(n_1, k_1) + (n_2, k_2) = (n_1 + n_2, k_1 + k_2)$

Scalar multiplication in $\mathbb{Q}_\delta$: $c(n, k) = (cn, ck)$

Comparison in $\mathbb{Q}_\delta$: $(n_1, k_1) \leq (n_2, k_2)$ iff $(n_1 < n_2)$ or $(n_1 = n_2$ and $k_1 \leq k_2)$

Strict inequalities

So far we have used bounds like $x \leq 3$, but never strict ones like $x < 3$

Since we are on the rationals $x < 3$ is same as $x \leq (3 - \delta)$, where δ is a symbolic positive parameter $(\delta > 0)$

We do the following: instead of using $\mathbb{Q}$ numbers of form n, we use $\mathbb{Q}_\delta$ numbers of form (n, k) to be intended as $n + k\delta$

Addition in $\mathbb{Q}_\delta$: $(n_1, k_1) + (n_2, k_2) = (n_1 + n_2, k_1 + k_2)$

Scalar multiplication in $\mathbb{Q}_\delta$: $c(n, k) = (cn, ck)$

Comparison in $\mathbb{Q}_\delta$: $(n_1, k_1) \leq (n_2, k_2)$ iff $(n_1 < n_2)$ or $(n_1 = n_2$ and $k_1 \leq k_2)$

If constraints are satisfiable, it is always possible to compute a rational value for δ to translate $\mathbb{Q}_\delta$ numbers into $\mathbb{Q}$ numbers

Strict inequalities

Example: situation after receiving
$x_3 < -2$, $x_4 > 0$
and after $Update(x_3, (-2, -1))$, $Update(x_4, (0, 1))$

Tableau			lb		Bounds		ub	μ		
x_1	$=$	$-x_3 + x_4$	$-\infty$	$\leq$	x_1	$\leq$	∞	x_1	$\mapsto$	$(2, 2)$
x_2	$=$	$x_3 + x_4$	$-\infty$	$\leq$	x_2	$\leq$	∞	x_2	$\mapsto$	$(-2, 0)$
			$-\infty$	$\leq$	x_3	$\leq$	$(-2, -1)$	x_3	$\mapsto$	$(-2, -1)$
			$(0, 1)$	$\leq$	x_4	$\leq$	∞	x_4	$\mapsto$	$(0, 1)$

$x_1 = -(-2, -1) + (0, 1) = (2, 1) + (0, 1) = (2, 2)$
$x_2 = (-2, -1) + (0, 1) = (-2, 0)$

Solving $\mathcal{LIA}$ with $\mathcal{LRA}$

The Simplex can be used also to reason (in a complete way) about the integers ($\mathcal{LIA}$), e.g., using known techniques in linear programming

Given a set of $\mathcal{LIA}$ constraints S

- If S is unsatisfiable on $\mathcal{LRA}$[1] then it is also unsatisfiable on $\mathcal{LIA}$
- If S is satisfiable on $\mathcal{LRA}$, then we have to check if there is an integer solution

[1] This means that we allow variables to assume values in $\mathbb{Q}$ instead of $\mathbb{Z}$.

Solving $\mathcal{LIA}$ with $\mathcal{LRA}$

The Simplex can be used also to reason (in a complete way) about the integers ($\mathcal{LIA}$), e.g., using known techniques in linear programming

Given a set of $\mathcal{LIA}$ constraints S

- If S is unsatisfiable on $\mathcal{LRA}$[1] then it is also unsatisfiable on $\mathcal{LIA}$
- If S is satisfiable on $\mathcal{LRA}$, then we have to check if there is an integer solution

For the latter case the convex polytope on $\mathbb{Q}$ is explored sistematically. However, in general, search is necessary: $\mathcal{LIA}$ is NP-Complete, like SAT

- in SAT we split a and $\neg a$, in $\mathcal{LIA}$ we split $x \leq c$ and $x \geq c + 1$
- in SAT we learn clauses, in $\mathcal{LIA}$ we learn new tableau rows (new constraints)

[1] This means that we allow variables to assume values in $\mathbb{Q}$ instead of $\mathbb{Z}$.

Solving $\mathcal{LIA}$ with $\mathcal{LRA}$

The Simplex can be used also to reason (in a complete way) about the integers ($\mathcal{LIA}$), e.g., using known techniques in linear programming

Given a set of $\mathcal{LIA}$ constraints S

- If S is unsatisfiable on $\mathcal{LRA}$[1] then it is also unsatisfiable on $\mathcal{LIA}$
- If S is satisfiable on $\mathcal{LRA}$, then we have to check if there is an integer solution

For the latter case the convex polytope on $\mathbb{Q}$ is explored sistematically. However, in general, search is necessary: $\mathcal{LIA}$ is NP-Complete, like SAT

- in SAT we split a and $\neg a$, in $\mathcal{LIA}$ we split $x \leq c$ and $x \geq c + 1$
- in SAT we learn clauses, in $\mathcal{LIA}$ we learn new tableau rows (new constraints)

When on the integers, δ is set to 1 ($\mathbb{Q}_\delta$ numbers are not used)

[1] This means that we allow variables to assume values in $\mathbb{Q}$ instead of $\mathbb{Z}$.

Exercizes

1. Show that the $\mathcal{T}$-conflicts generated by the Simplex are minimal
2. Suppose that $(0, 1) \leq x \leq (1, -1)$. Compute the biggest possible value for δ
3. Find the candidate for pivoting in this row (for simplicity we use normal numbers)

$$x_1 = 3x_2 - 9x_3 - 7x_4$$

   given these bounds $-\infty \leq x_1 \leq 1$, $0 \leq x_2 \leq \infty$, $-\infty \leq x_3 \leq -1$, $1 \leq x_4 \leq 2$ and this assignment
   $\mu = \{x_1 \mapsto 2, x_2 \mapsto 0, x_3 \mapsto -1, x_4 \mapsto 1\}$
4. Using the row of previous exercize, change the bounds so that the only candidate for pivoting becomes x_2
5. Now change the bounds so that no pivoting is possible. Compute the conflict