# Decidable Linear List Constraints

Sabine Bauer and Martin Hofmann

LPAR 21, May 9, 2017

LUDWIG-
MAXIMILIANS-
UNIVERSITÄT
MÜNCHEN

# Problem Definition: Constraints on Infinite Lists

- Linear constraints with integer coefficients (as $\mathbf{x} + \mathbf{y} \leq 3\mathbf{z} + \mathbf{w}$) with variables ranging over infinite lists of nonnegative rational numbers
- Addition and comparison understood pointwise
- lists can be shifted (**tl** operation)
- In addition arithmetic inequalities between selected components of lists

## Asked

(Simultaneous) satisfiability (of all list and arithmetic constraints) with list entries in $\mathbb{Q}_0^+$?

# Examples

$\mathrm{tl}(\mathrm{tl}(x))$ becomes $\mathrm{tl}^2(x)$

$\mathbf{tl}^2(\mathbf{x}) \geq \mathbf{tl}(\mathbf{x}) + \mathbf{x}$

$$
\begin{array}{l}
1, 1, 2, 3, 5, 8, 13, \ldots + \\
1, 2, 3, 5, 8, 13, 21, \ldots = \\
2, 3, 5, 8, 13, 21, 34, \ldots
\end{array}
$$

Grammar

$$
\begin{aligned}
\mathit{list} &::= \mathit{variable} | \mathit{tl}(\mathit{list}) \\
\mathit{sum} &::= \mathit{list} | \mathit{sum} + \mathit{sum} \\
\mathit{constraint} &::= \mathit{sum} \geq \mathit{sum}
\end{aligned}
$$

Thus $\mathbf{x} = 1, 1, 2, 3, 5, 8, 13, \ldots$ satisfies $\mathbf{tl}^2(\mathbf{x}) \geq \mathbf{tl}(\mathbf{x}) + \mathbf{x}$. So does for example also $\mathbf{x} = 1, 1, 100, 101, 201, 310, \ldots$

Each solution list is pointwise greater or equal to a Fibonacci list.

# Satisfiable Example

The following constraint system is satisfiable

$$\mathbf{tl}^2(\mathbf{x}) + \mathbf{y} \geq \mathbf{tl}(\mathbf{x}) + \mathbf{x},$$
$$\mathbf{hd}(\mathbf{x}) = 1 \geq \mathbf{hd}(\mathbf{tl}(\mathbf{x})).$$

and has for example the solutions $\mathbf{y} = 0, 0, 0, ....$ and $\mathbf{x} = 1, 1, 2, 3, 5, 8, ....$

# Another Satisfiable Example

| Arithmetic Constraints | List Constraints |
|---|---|
| $\mathbf{hd}(\mathbf{x}) = \mathbf{hd}(\mathbf{z}) = 2$ | $\mathbf{tl}(\mathbf{y}) \geq \mathbf{y}$ |
| $\mathbf{hd}(\mathbf{tl}(\mathbf{x})) = 5$ | $\mathbf{z} \geq \mathbf{tl}(\mathbf{z}) + \mathbf{tl}(\mathbf{z})$ |
| $\mathbf{hd}(\mathbf{y}) \geq 1$ | $\mathbf{tl}(\mathbf{tl}(\mathbf{x})) \geq \mathbf{tl}(\mathbf{x}) + \mathbf{x} + \mathbf{tl}(\mathbf{y}) + \mathbf{tl}(\mathbf{y})$ |

The same example in array notation:

| Arithmetic Constraints | List Constraints |
|---|---|
| $\mathbf{x}[0] = \mathbf{z}[0] = 2$ | $\mathbf{y}[i+1] \geq \mathbf{y}[i] \forall i \geq 0$ |
| $\mathbf{x}[1] = 5$ | $\mathbf{z}[i] \geq \mathbf{z}[i+1] + \mathbf{z}[i+1] \forall i \geq 0$ |
| $\mathbf{y}[0] \geq 1$ | $\mathbf{x}[i+2] \geq \mathbf{x}[i+1] + \mathbf{x}[i] + \mathbf{y}[i+1] + \mathbf{y}[i+1] \forall i \geq 0$ |

Solutions: $\mathbf{y} = 1, 1, 1, ..., \mathbf{z} = 2, 1, 0.5, 0.25, 0.125, ...,$
$\mathbf{x} = 2, 5, 9, 16, 27, 45, x_n, x_{n+1}, x_n + x_{n+1} + 2, ...$

# Unsatisfiable Examples

1. $$\mathbf{hd}(\mathbf{x}) \leq 1 \wedge \mathbf{hd}(\mathbf{x}) \geq 2,$$

2. $$\begin{aligned} \mathbf{hd}(\mathbf{x}) \geq 1, \mathbf{tl}^{\mathbf{2}}(\mathbf{x}) &\geq \mathbf{x} + \mathbf{tl}(\mathbf{x}), \\ &\rightsquigarrow \mathbf{x} \geq 1, 0, 1, 1, 2, 3, 5, \ldots \\ \mathbf{hd}(\mathbf{tl}^{\mathbf{6}}(\mathbf{x})) \leq 4 \rightsquigarrow \text{UNSAT} \end{aligned}$$

3. $$\begin{aligned} \mathbf{hd}(\mathbf{x}) &= 1, \mathbf{tl}^{\mathbf{2}}(\mathbf{x}) \geq \mathbf{x} + \mathbf{tl}(\mathbf{x}), (\mathbf{x} \text{ as above }) \\ \mathbf{hd}^{\mathbf{31}}(\mathbf{z}) &= 2, \\ \mathbf{tl}^{\mathbf{5}}(\mathbf{z}) &\geq \mathbf{z} + \mathbf{u}, \\ \mathbf{z} &\geq \mathbf{tl}^{\mathbf{3}}(\mathbf{z}) + \mathbf{v}, \\ &\rightsquigarrow \mathbf{hd}(\mathbf{tl}^{\mathbf{5+15k}}(\mathbf{z})) = \mathbf{hd}(\mathbf{tl}^{\mathbf{5}}(\mathbf{z})) \forall k \in \mathbb{N}, \\ \mathbf{z} &\geq \mathbf{x} \rightsquigarrow \text{UNSAT} \end{aligned}$$

# Application to resource analysis

- MH and Rodriguez: resource analysis with type systems, in particular prediction of memory usage as a function of the input size
- In earlier LPAR H&R proposed list constraints (generalized to tree constraints) as a backend of the analysis, gave incomplete heuristic procedure to solve them
- General problem of deciding satisfiability for list/tree constraints remained open

# Black Box: Translation between Programs and Constraints

- Given: Java program (certain fragment: RAJA) with main function taking list of strings as input
- Asked: Bound on memory usage as function of input size
- Technique: amortized analysis with potential method
  - gives worst case average runtime for sequences of operations
  - takes into account how the data structures change during the computation

- Analysis defines potential of data structures using infinite lists (trees) which must be chosen so as to satisfy typing rules (potentials always add up)
- Type inference uses unknown lists (trees) to compile program into set of constraints on those unknowns

In this case the program can execute with an amount of memory that can be read off from the constraint solutions.

# Related Work

From now on we are in the realm of linear arithmetic on infinitely many variables.

- MH and Rodriguez: "Linear constraints on infinite trees" incomplete heuristic procedure for trees
- "On infinite CSPs" Dantchev&Valencia: index arithmetic (e.g.. for all even indices some predicate holds)
- "What else is decidable about infinite arrays?" Habermehl&Iosif&Vojnar: Constant differences, no sums

N.B.: lots of related work on resource analysis, distinctive feature of RAJA: allows a very wide range of resource behaviors, not restricted to for instance polynomials.

# Outline

For list constraint systems we provide the following answers:

1. The general problem as formulated by MH&Rodriguez hard for the (notoriously difficult) Skolem-Mahler-Lech problem
2. We analyzed the translation from resource inference $\rightsquigarrow$ list/tree constraints more closely and proved that it only produces a *proper subset* of the general problem. We identified this fragment and called it "unilateral"
3. Satisfaction of unilateral list constraints are shown decidable in polynomial time.
4. We can obtain upper bounds on minimal solutions $\rightsquigarrow$ good upper bounds on resource usage

# Hardness of the General List Case

- Since the constraint solutions form a convex set, the first thought was that it could be decidable by using methods of convex optimization
- The positivity problem for recurrences ("Is each value of a given linear recurrence nonnegative?") can be reduced to constraint satisfiability
- Encode initial values as linear program with unique solution
- Translation of a recurrence relation to a list constraint

$$\mathbf{x}_n = \frac{1}{2}\mathbf{x}_{n-1} - 2\mathbf{x}_{n-3}, n \geq 3$$

$$\Updownarrow$$

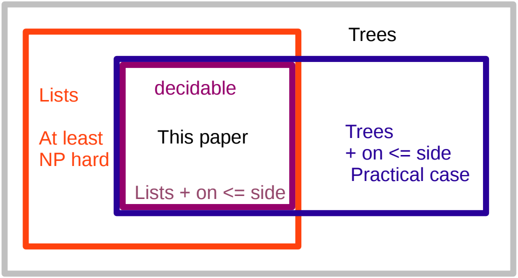
$$2\mathbf{x}_n + 4\mathbf{x}_{n-3} = \mathbf{x}_{n-1}, n \geq 3$$

$$\Updownarrow$$

$$\mathbf{tl}(\mathbf{tl}(\mathbf{tl}(\mathbf{x}))) + \mathbf{tl}(\mathbf{tl}(\mathbf{tl}(\mathbf{x}))) + \mathbf{x} + \mathbf{x} + \mathbf{x} + \mathbf{x} + \mathbf{x}$$
$$\geq \text{(and } \leq) \mathbf{tl}^2(\mathbf{x})$$

- Special subcase: Which constraints really appear in practice?

# Unilateral Constraints




- Special case covering all instances that stem from amortized analysis: $+$ only on $\leq$ side of the inequality.
- Examples:
  - $\mathbf{tl}^1(\mathbf{x}) \geq \mathbf{tl}^4(\mathbf{x}) + \mathbf{tl}^8(\mathbf{x}) + \mathbf{tl}^8(\mathbf{x})$ is in this form
  - $\mathbf{tl}^8(\mathbf{x}) + \mathbf{tl}^4(\mathbf{x}) \geq \mathbf{tl}^1(\mathbf{x})$ is not

# Unilateral List Constraint Satisfiability is Decidable: Proof Idea

Unilateral constraint system

$\downarrow$

Equisatisfiable unilateral constraint system with periodic solutions

$\downarrow$

Linear program

# Reduction to the periodic case

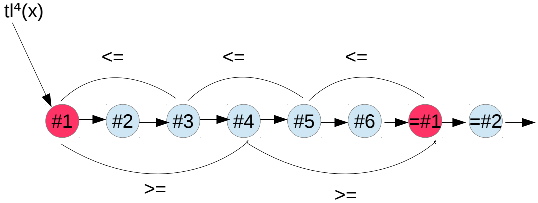
Given: Constraints in *normal form*: basically, all variables appear on two sides of an inequality (The rest can be removed)

## Bounds in one direction

From an finite position on, we can set all zero or infinity
(e.g. $\mathbf{tl}^4\mathbf{x} \geq \mathbf{tl}^2\mathbf{x} \to \mathbf{x} = \infty, \infty, \ldots$ or $\mathbf{tl}^4\mathbf{x} \geq \mathbf{tl}^7\mathbf{x} \to \mathbf{x} = 0, 0, \ldots$)

## Bounds in two directions imply periodicity

$\mathbf{tl}^4\mathbf{x} \geq \mathbf{tl}^2\mathbf{x} + \mathbf{tl}^7\mathbf{x}$
$\Rightarrow \mathbf{tl}^4\mathbf{x} \geq \mathbf{tl}^2\mathbf{x}$, in each second position the list entries are nondecreasing
$\Rightarrow \mathbf{tl}^4\mathbf{x} \geq \mathbf{tl}^7\mathbf{x}$, in each third position the list entries nonincreasing



# Translation of List Constraints to a Linear Program

$$\mathbf{x} \geq \mathbf{y}_1 + \cdots + \mathbf{y}_k,$$

has the equivalent formulation (for x,y periodic with period length 3)

$$\begin{aligned} x_0 x_1 \dots x_m (x'_1 x'_2 x'_3)^\omega \geq & y_{1,0} y_{1,1} \dots y_{1,m_1} (y'_{1,1} y'_{1,2} y'_{1,3})^\omega + \dots \\ & + y_{k,0} y_{k,1} \dots y_{k,m_k} (y'_{k,1} y'_{k,2} y'_{k,3})^\omega \end{aligned}$$

We add an arithmetic constraint

$$\begin{aligned} \mathbf{hd}(x_0 x_1 \dots x_m (x'_1 x'_2 x'_3)^\omega \geq & \mathbf{hd}(y_{1,0} y_{1,1} \dots y_{1,m_1} (y'_{1,1} y'_{1,2} y'_{1,3})^\omega) + \\ & + \mathbf{hd}(y_{k,0} y_{k,1} \dots y_{k,m_k} (y'_{k,1} y'_{k,2} y'_{k,3})^\omega), \end{aligned}$$

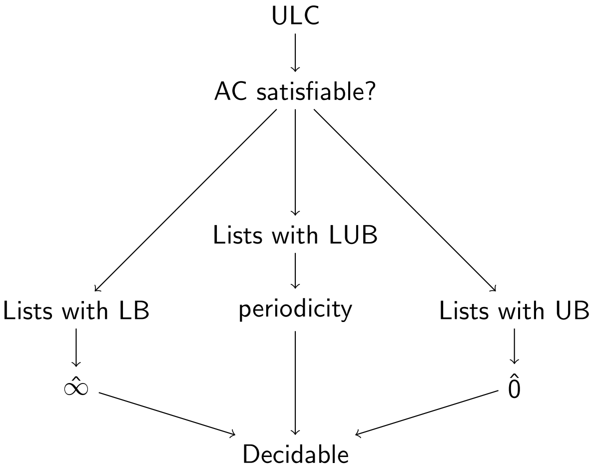
and then continue recursively with **tl**

$$\begin{aligned} x_1 \dots x_m (x'_1 x'_2 x'_3)^\omega \geq & y_{1,1} \dots y_{1,m_1} (y'_{1,1} y'_{1,2} y'_{1,3})^\omega + \dots \\ & + y_{k,1} \dots y_{k,m_k} (y'_{k,1} y'_{k,2} y'_{k,3})^\omega. \end{aligned}$$

This procedure stops after a finite number of steps.

# Decidability for the special list case

Let ULC be a unilateral system of arithmetic constraints AC and list constraints, partitioned in the three sets LB (lower bounds), UB (upper bounds) and LUB (lower and upper bounds). Then the satisfiability question for ULC is decidable.



# Optimal solutions

- Optimal solutions = minimal solutions (most accurate upper bounds on resource usage)
- In the general case: add constraints that ensure polynomial/exponential growth and check if still satisfiable
- Such upper bounds do not fall into the unilateral fragment
- Upper bounds for unilateral constraint solutions are nonincreasing

# Growth Behavior I: From Constraints to Matrices

- The problem can be written as a matrix exponentiation problem
- We can set up a matrix for each part of the system where there is only one constraint on each variable.

## Example

$$\mathbf{x}^{(n)} \geq 3\mathbf{x}^{(n-2)} + 4\mathbf{y}^{(n-1)},$$

$$\mathbf{x}^{(n)} \geq \mathbf{x}^{(n-1)} + 2\mathbf{x}^{(n-2)} + 4\mathbf{y}^{(n-1)},$$

$$\mathbf{y}^{(n)} \geq 3\mathbf{x}^{(n-1)} + \mathbf{y}^{(n-1)}.$$

becomes

$$\begin{pmatrix} \mathbf{x}^{(n)} \\ \mathbf{x}^{(n-1)} \\ \mathbf{y}^{(n)} \end{pmatrix} \geq \begin{pmatrix} 0 & 3 & 4 \\ 1 & 0 & 0 \\ 3 & 0 & 1 \end{pmatrix} \begin{pmatrix} \mathbf{x}^{(n-1)} \\ \mathbf{x}^{(n-2)} \\ \mathbf{y}^{(n-1)} \end{pmatrix},$$

$$\begin{pmatrix} \mathbf{x}^{(n)} \\ \mathbf{x}^{(n-1)} \\ \mathbf{y}^{(n)} \end{pmatrix} \geq \begin{pmatrix} 1 & 2 & 4 \\ 1 & 0 & 0 \\ 3 & 0 & 1 \end{pmatrix} \begin{pmatrix} \mathbf{x}^{(n-1)} \\ \mathbf{x}^{(n-2)} \\ \mathbf{y}^{(n-1)} \end{pmatrix}.$$

# Growth Behavior II: One Constraint per Variable

## Theorem

*Let $\mathbb{L}$ be the set of solutions for a particular variable in a list constraint system with one constraint per variable. We can effectively find (in polynomial time) $k \in \mathbb{N}$ and $r \geq 0$ and $c > 0$ such that*

- *for all $\mathbf{x} \in \mathbb{L}$*

$$\mathbf{x}(n) \geq cn^k r^n \text{ for infinitely many } n.$$

- *there exist a constant $c' \in \mathbb{R}$ and $\mathbf{x} \in \mathbb{L}$ with*

$$\mathbf{x}(n) \leq c' n^k r^n.$$

*Thus the asymptotic growth of the minimal solution is $n^k r^n$.*

# Growth Behavior III: Two Constraints for one Variable

General recursive relation:

$$\mathbf{x}_n = \begin{pmatrix} \max((A_1\mathbf{x}_{n-1})_1, ..., (A_l\mathbf{x}_{n-1})_1) \\ ... \\ \max((A_1\mathbf{x}_{n-1})_m, ..., (A_l\mathbf{x}_{n-1})_m) \end{pmatrix}$$

Example (two constraints on first variable, one on the second):

$$\mathbf{x}_n \geq 2(3)\mathbf{x}_{n-1} + 3(2)\mathbf{y}_{n-1} \wedge \mathbf{y}_n \geq 4\mathbf{x}_{n-1} + 1\mathbf{y}_{n-1},$$

$$A = \begin{pmatrix} 2 & 3 \\ 4 & 1 \end{pmatrix} \text{ and } B = \begin{pmatrix} 3 & 2 \\ 4 & 1 \end{pmatrix}, \mathbf{x}_n = \begin{pmatrix} \max((A\mathbf{x}_{n-1})_1, (B\mathbf{x}_{n-1})_1) \\ (A\mathbf{x}_{n-1})_2 \end{pmatrix}$$

We have for all $v \neq (a, a)$

$$Av > Bv \Rightarrow BAv > AAv \wedge Av < Bv \Rightarrow ABv > BBv.$$

Chain of "better" matrices is alternating: ABABAB...

# Upper Bound on Minimal Growth

This chains of better matrices seem to be aperiodic in some cases.

## Approximation

We replace the $k$ constraints on $\mathbf{x}$ with coefficients $a_{i,j}$ where $i = 1, ...k$ and $j = 1, ..., m$, with $m = max(\dim A_i)$ by the new constraint

$$\mathbf{x}_n \geq \max(a_{1,1}, \ldots a_{k,1})\mathbf{x}_{n-1} + \cdots + \max(a_{1,m}, \ldots, a_{k,m})\mathbf{x}_{n-m}.$$

This still gives a nontrivial upper bound on the minimal solution.

# Summary and Conclusion

1. Theory: We have proven decidability for a problem involving linear arithmetic of infinite lists.
2. Application:
   - Automatic type inference for list programs is now possible by a reduction to linear programming.
   - We can give nontrivial upper bounds on the resource consumption.
   - We have started implementing the decision procedure and adapting it to the RAJA tool.
3. Future Work
   - Decidability of the tree case (or undecidability)
   - Is LC (general constraint satisfiability) equivalent to SML or even harder?