Arithmetical Module for MetaPRL: rules and Arith tactic

Yegor Bryukhov

GC CUNY

ybryukhov@gc.cuny.edu

Name translation

- Integers is a base type, by e□ciency reasons it is not emulated by recursive structures
- Each integer number has it's own name
- Computations on constants has to be implemented as ML-rules

Examples of rules

$$\frac{\Gamma \vdash a \in Z \qquad \Gamma \vdash b \in Z}{\Gamma \vdash (a <_b b) \wedge_b (b <_b a) \Rightarrow \mathrm{ff}} \qquad ($$

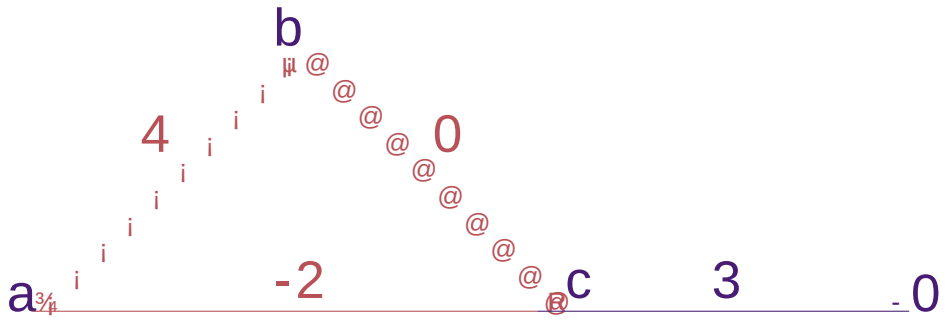
Brief overview of arithmetic rules

- Computations on constants
- Types
- wf-rules
- Ring axioms
- Discrete linear order of $<$
- Definitions of other relations ($>; \cdot; \neg$)
- Monotonicity axioms (correlation/F519.83Tf30perations and relation

Arith algorithm

- Move negated conclusion to assumptions
- Rewrite all relations to $\neg$
- Rewrite all polynomials to canonical form
- Provide each relation is of form $P \neg Q + C$, where $P(0) = Q(0) = 0$
- $P \neg Q + C$ is an edge from P to Q with weight C
- Find a positive cycle
- Sum relations corresponding to this cycle
- Conclude that $P \neg P + C$, $C > 0$ is false

Example: Graph



Limitations of Arith

Arith can't prove the following:

$$\frac{\Gamma \vdash a \in \mathbb{Z} \quad \Gamma \vdash b \in \mathbb{Z} \quad \Gamma \vdash c \in \mathbb{Z} \quad \Gamma \vdash a < b}{\Gamma \vdash (a + c) < (b + c}$$

What has been done

- Move negated conclusion to assumptions
- Rewrite all relattons to $\neg$
- Rewrite all polynomials to canonical form
- Provide each relation is of form $P \neg Q + C$, where $P(0) = Q(0) = 0$
- $P \neg Q + C$ is an edge from P to Q with weight C

Arith for other logics

- Graph algorithm is a separate module
- “relations to ¸” should be implemented in resource-based way

Towards SupInf

- Less explicit result
- Reasoning in rational numbers

- Rational Numbers Theory
- Formalization of SupInf