

ReFuncTion

Conditional Termination by Abstract Interpretation of Numerical C Programs

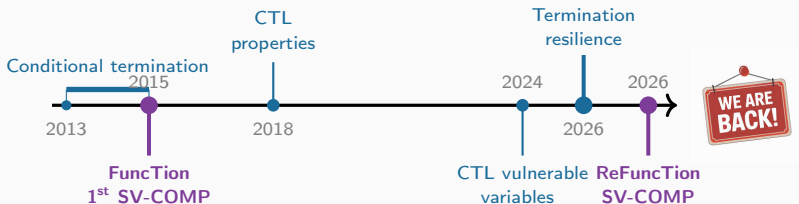
Competition Contribution — SV-COMP 2026

Naïm Moussaoui Remil Caterina Urban

Inria & ENS | PSL, Paris, France

What is ReFuncTion?

A **sound static analyser** for numerical C programs, built on abstract interpretation and forked from FuncTion.

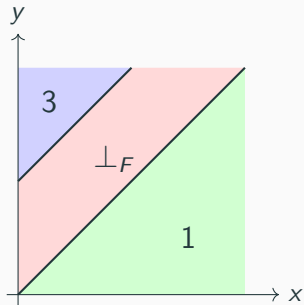
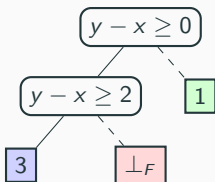


What's new in ReFuncTion

- New parser (Mopsa)
- Signed machine integers
- Interprocedural analysis

Decision Tree Abstract Domain

Internal **nodes** are linear constraints partitioning the state space; **leaves** are affine ranking functions (or $\perp_F$ when termination is not proved).



ReFuncTion infers *piecewise ranking functions* via a backward analysis.
Over 2 384 tasks it remained **sound** — zero incorrect answers.

Score

Correct:	<u>111</u>
Unknown:	131
Timeouts:	206
Unsupported:	<u>1 926</u>

Strengths

Sound; fast on supported programs; infers termination preconditions.

Weaknesses

Narrow language support; usage of a naïve widening.

Perspectives — Future Editions

Two axes: extend *what* ReFuncTion can verify, and *which* programs it handles.

Property scope (CTL)

- Reachability / safety
- Non-termination & bug witnesses
- More

Program scope

- Heap-manipulating programs
- Unsigned integers, recursion
- goto / break



<https://github.com/naim-mr/refunction>