Goose: A Meta-Solver for Deep Neural Network Verification

Reporter: Chi Zhiming

November 18, 2022

目录

目录

Contribution

- Goose, a meta-solver, leverages three key meta-solving techniques, namely, adaptive algorithm selection, probabilistic satisfiability inference, and time interval deepening to implement an adaptive sequential portfolio of solvers for NN verification.
- Goose improve 37.7% across benchmarks and solvers from VNN-COMP'21 and 41.4% over competition solvers on a 90 select ACAS Xu instances .

Background

VNN-COMP

- International Verification of Neural Networks Competition
- website link

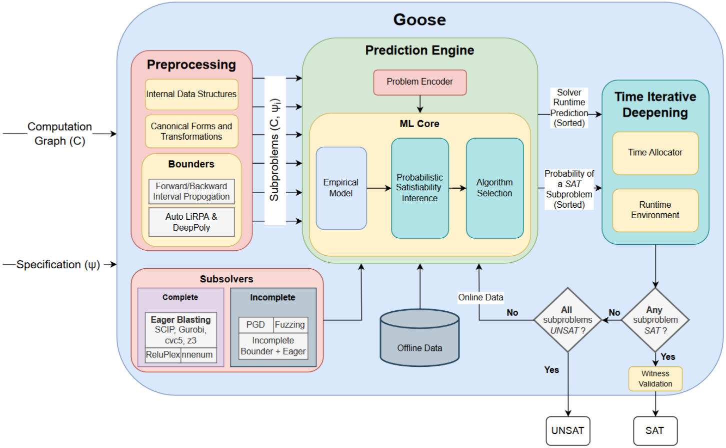
Total Score

#	Tool	Score
1	α,β-CROWN	1274.9
2	MN BaB	1017.3
3	Verinet	892.5
4	Nnenum	534.0
5	Cgdtest	406.4
6	Peregrinn	399.0
7	Marabou	380.6
8	Debona	222.9
9	Fastbatllnn	100.0
10	Verapak	98.2
11	Averinn	29.1

GPU-based

目录

Framework


Goose
Computation Graph (C)
Specification (ψ)
Preprocessing
Internal Data Structures
Canonical Forms and Transformations
Bounders
Forward/Backward Interval Propogation
Auto LiRPA & DeepPoly
Subproblems (C, ψ_i)
Prediction Engine
Problem Encoder
ML Core
Empirical Model
Probabilistic Satisfiability Inference
Algorithm Selection
Solver Runtime Prediction (Sorted)
Probability of a SAT Subproblem (Sorted)
Time Iterative Deepening
Time Allocator
Runtime Environment
Subsolvers
Complete
Eager Blasting
SCIP, Gurobi, cvc5, z3
ReluPlex
nnenum
Incomplete
PGD
Fuzzing
Incomplete Bounder + Eager
Offline Data
Online Data
No
All subproblems UNSAT ?
No
Any subproblem SAT ?
Yes
Yes
Witness Validation
UNSAT
SAT

Preliminary Study

Problem Encoder:

- Goose implements a problem encoder $\xi\left(C;\psi_i\right)$ to compute **feature vectors** - a real-valued vector representation of C, ψ_i, with 221 dimensions (features).
- Network, specification, online information and subproblem features.

Probabilistic Satisfiability Inference

- determine which subproblems to be targeted using decision NN model.

Algorithm selection:

- determine which solvers should be used decision NN model.

Pseudocode for Goose

Algorithm 1 The main execution loop of `Goose`

Input: A computation graph C and a linear specification ψ over C
Output: `SAT/UNSAT`

procedure Goose-MainLoop
  $t = t_{init}$
  $\mathcal{P} = \mathcal{T}(C, \psi)$ ▷ $\mathcal{T}$ is the transform from Theorem 1
  $\mathcal{F} = [\bot \ \forall C, \psi_i \ \in \mathcal{P}]$ ▷ subproblem flag
  solved $= \bot$
  while not solved **do**
    α_{ψ_i} = allocation of t for each unsolved ψ_i from probabilistic satisfiability inference.
    Sort $\mathcal{P}$ in descending order by α_{ψ_i}
    for C, ψ_i in $\mathcal{P}$ **do**
      if $\mathcal{F}[C, \psi_i]$ **then**
        continue
      end if
      β_{s,ψ_i} = allocation of α_{ψ_i} for each $s \in \mathcal{S}$ from algorithm selection.
      Sort $\mathcal{S}$ in descending order by β_{s,ψ_i}
      for $s \in \mathcal{S}$ **do**
        $\rho = \mathtt{run}(s, C, \psi_i, \beta_{s,\psi_i})$
        if ρ is `SAT` **then** ▷ If any subproblem ψ_i is `SAT`
          return SAT
        else if ρ is `UNSAT` **then**
          $\mathcal{F}[C, \psi_i] = \top$
        end if
      end for
    end for
    t += an exponential increment
    solved $= \bigwedge_{v \in \mathcal{F}} v$
  end while
  return `UNSAT`
end procedure

目录

benchmark of VNN-COMP'21

The PAR-2 score of a solver on a benchmark is the wallclock runtime if successful, otherwise twice the wallclock runtime (lower is better).

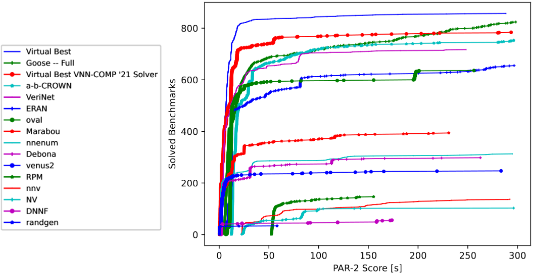


Fig. 2. Main experimental CDF plot over VNN-COMP '21 benchmarks (Section 4.2). A CDF is a visualization of a solver's performance on a benchmark suite the vertical axis represents the number of benchmarks solved (higher is better) and the horizontal axis is the benchmark wise `PAR-2` (lower is better). See abalation study in

benchmark of Acasxu

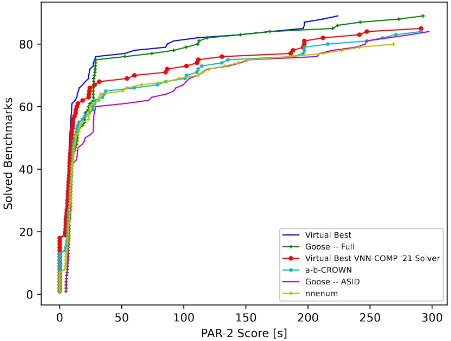


Fig. 3. CDF plot over select ACAS Xu benchmarks (Section 4.3). A CDF plot is a visualization of a solver's performance on a benchmark suite the horizontal axis represents the number of benchmarks solved (higher is better) and the vertical axis is the benchmark wise `PAR-2` (lower is better).

Thank you