FCC LEARN Program

Federal Communications Commission

Part I:

Role of Feasibility Checking in the Reverse Auction

Brett Tarnutzer
Assistant Bureau Chief, WTB

brett.tarnutzer@fcc.gov

February 21, 2014

Recap of Relevant Releases

- Updated TVStudy software based on OET-69
- Data Public Notice
  - Technical Appendix describing constraint generation
  - Constraint files:
    - Domain File (*Domain_2013July15.csv*)
    - Interference _Paired File (*Interference_Paired_2013July15.csv*)

Domain File

- Considering fixed constraints, the domain file provides a list of possible channels each station could be assigned in the repacking process

Station ID | **Available Channels**

```
DOMAIN, 10001, 2, 3, 4, 5, 6, 19, 20, 21, 48, 49, 50, 51
DOMAIN, 10002, 2, 3, 4, 5, 6, 7, 8, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 38, 39, 40
DOMAIN, 10003, 2, 3, 4, 5, 6, 7, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 34, 35, 40, 41, 42, 43, 44, 45
```

Interference Constraint File

- Considers interference between pairs of TV stations on co- or adjacent-channels
- For a given station, lists all the other stations that reduce its baseline interference-free population by more than a specific amount

Interference Type

Band Limits

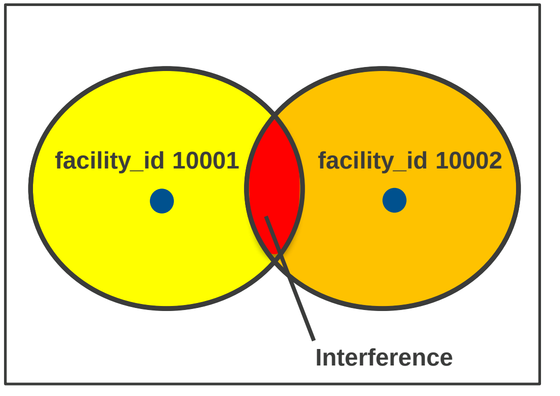
Study Station

Interfering Stations

```
CO,     2, 4, 10036, 10057, 10066, 10118, 10282, 10345, 10438
ADJ+1, 2, 4, 10913, 10281, 10761, 10864
ADJ-1, 2, 4, 10442, 10037, 10675
```

How The FCC Can Use The Data

- In the context of the reverse auction:
  - Used in determining which bids can be accepted, based on the feasibility of being able to assign the station a channel in its home band
  - For stations that remain on-air, ensures at least one feasible channel assignment exists

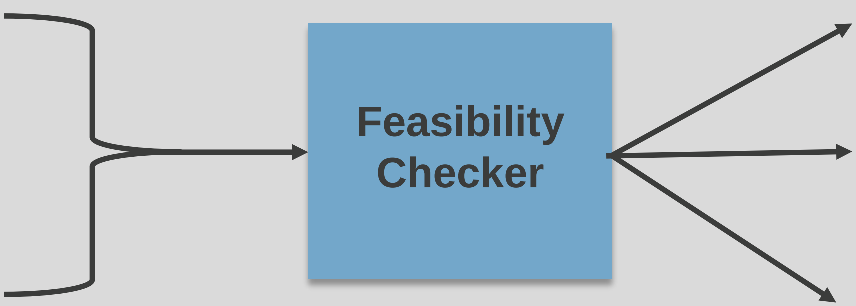


The Feasibility Question

Can a given set of TV stations be assigned a channel in a particular band such that none of the interference constraints are violated?

Key Inputs

1) Question file
2) Domain file
3) Interference file



Outputs (in Answer File)

- **Yes** (assignment of stations to channels)
- **No**
- **Unknown** (in time allotted)

Multiple Round Auction with Feasibility Checking

- As prices decline and a station is unwilling to accept a given price, the station will be assigned a channel in its home band
- At this point, each active station must be checked for a feasible assignment to their home band
  - If the active station cannot be assigned a channel in its home band, its offer price is not further reduced for this clearing target
  - Otherwise, it remains active and the offer price will be reduced in the next round

Testing the Feasibility Checker

- Speed
  - How quickly can a feasibility check be performed in a multi-round auction?
- Certainty
  - Accurate solutions are vital to the repacking process
- Feasibility question generator
  - Rank order station by randomized bid amount
  - The station with the highest bid amount is selected to stay in its home band, all other stations must be checked for feasibility
  - Iterates

Possible Approaches to Answering the Feasibility Question

- Integer Optimization Solvers
- Constraint Programming Solvers
- Satisfiability Solvers

Federal Communications Commission

Part II:

Methodologies & Test Results

Dr. Kevin Leyton-Brown
Affiliate, Auctionomics;
Associate Prof. of Computer Science,
University of British Columbia

kevinlb@cs.ubc.ca

Satisfiability

- One of the most widely studied combinatorial optimization problems in computer science
  - Asks whether any truth assignment to a set of Boolean variables causes a given formula to evaluate to true
- Used in practice to solve many hard yes/no problems:
  - Circuit verification
  - Detecting bugs in software
  - Planning
  - Scheduling

Defining the Satisfiability Problem

- A (Boolean) **variable** is denoted $x_{i,,j}$, and can take the value true or false.
- A **literal** is a possibly negated variable, denoted $x_{i,,j}$ or $\neg x_{i,,j}$. The literal $\neg x_{i,,j}$ evaluates to true if $x_{i,,j}$ is false, and to false otherwise.
- A **clause** is a disjunction of literals: a list of literals connected by the OR operator, which is denoted by the symbol ∨. The clause $(x_{i,,j} \vee x_{k,,l})$ evaluates to false if $x_{i,,j}$ and $x_{k,,l}$ are both false, and to true otherwise.

Defining the Satisfiability Problem

- A **formula** is a conjunction of the whole set of clauses—that is, a list of all of the clauses, connected by the AND operator, which is denoted by the symbol ∧.
- If the set of clauses is {*C1*, *C2*, *C3*}, then the formula is *C1* ∧ *C2* ∧ *C3*. Given a truth assignment to the variables, this formula evaluates to true if each of *C1*, *C2* and *C3* evaluate to true, and to false otherwise.
- ***Does there exist any truth assignment to the variables that makes the formula evaluate to true?***

Solving Satisfiability Problems

- Satisfiability questions can be difficult to answer
  - In fact, NP-complete: in a formal sense, the hardest SAT problems are just as hard as the hardest instances of a wide range of other combinatorial optimization problems
- Why should it be hard to find a satisfying assignment?
  - Different clauses can contain the same variables, in some cases negated and in some cases not
  - Values for these variables must be chosen carefully so that each clause evaluates to true
- The good news: it's easy to **verify** a satisfiable assignment

SATFC

- Our strategy for TV station feasibility checking:
  - Encode the feasibility checking problem as a SAT problem
  - Run presolvers to weed out easy problems quickly
  - Run a specially-configured SAT solver on what remains
- Lots of engineering effort required to reduce overheads and make this fast in practice
- Thanks in particular to Alexandre Frechette for his efforts in coding and testing SATFC

SATFC: Encoding

- TV station feasibility checking can be encoded as a satisfiability problem
  - Decision variables define allowable channel assignments
  - Clauses enforce interference restrictions

- Variables:

$$x_{s,c} = \begin{cases} \text{true if station } s \epsilon S \text{ is assigned to channel } c \epsilon C_s \\ \text{false otherwise} \end{cases}$$

SATFC: Encoding: Clauses

For every pair of channels *c1* and *c2* allowed for station *s*, at most one can be assigned:

$(\neg x_{s,c1} \vee \neg x_{s,c2})$

Each station must take one of its allowable channels:

$(x_{s,c1} \vee \ldots \vee x_{s,cn})$

We must respect every pairwise interference rule given in `interference_paired.csv`, specifying that station *s1* cannot broadcast on channel *c1* while station *s2* broadcasts on channel *c2:*

$(\neg x_{s1,c1} \vee \neg x_{s2,c2})$

SATFC: Presolving

Ladder setting: we know that a given set of stations were packable, and face the question of whether one new station can be added

- **Test for unsat**
  - Drop all stations outside the neighborhood of the new station, solve the remaining problem
  - If the answer is unsat, the whole problem is unsat
- **Test for sat based on a previously satisfiable solution**
  - Force all non-neighboring stations to previous values, solve the remaining problem
  - If the answer is sat, the whole problem is sat

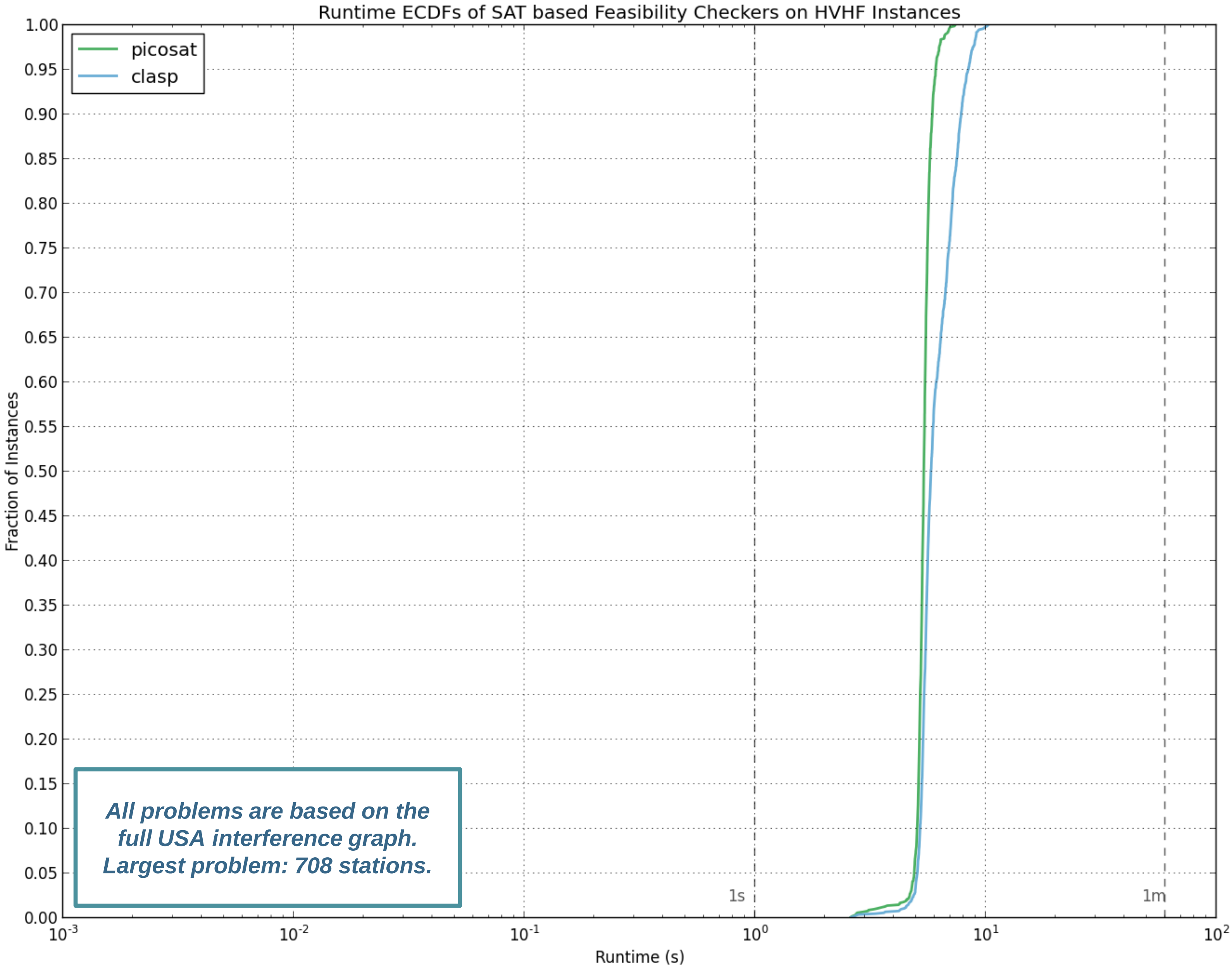
SATFC: Algorithm Configuration

- Artificial intelligence techniques for **automatically optimizing parameterized algorithms** for particular problems
  - One of my research group's key foci over the past decade
- For the station repacking problem, we used a technique called Sequential Model-based Algorithm Configuration (**SMAC**) to optimize the performance of a SAT solver
  - Configured separately for VHF and UHF problems
- Key SMAC collaborators: Frank Hutter, Holger Hoos, Steve Ramage

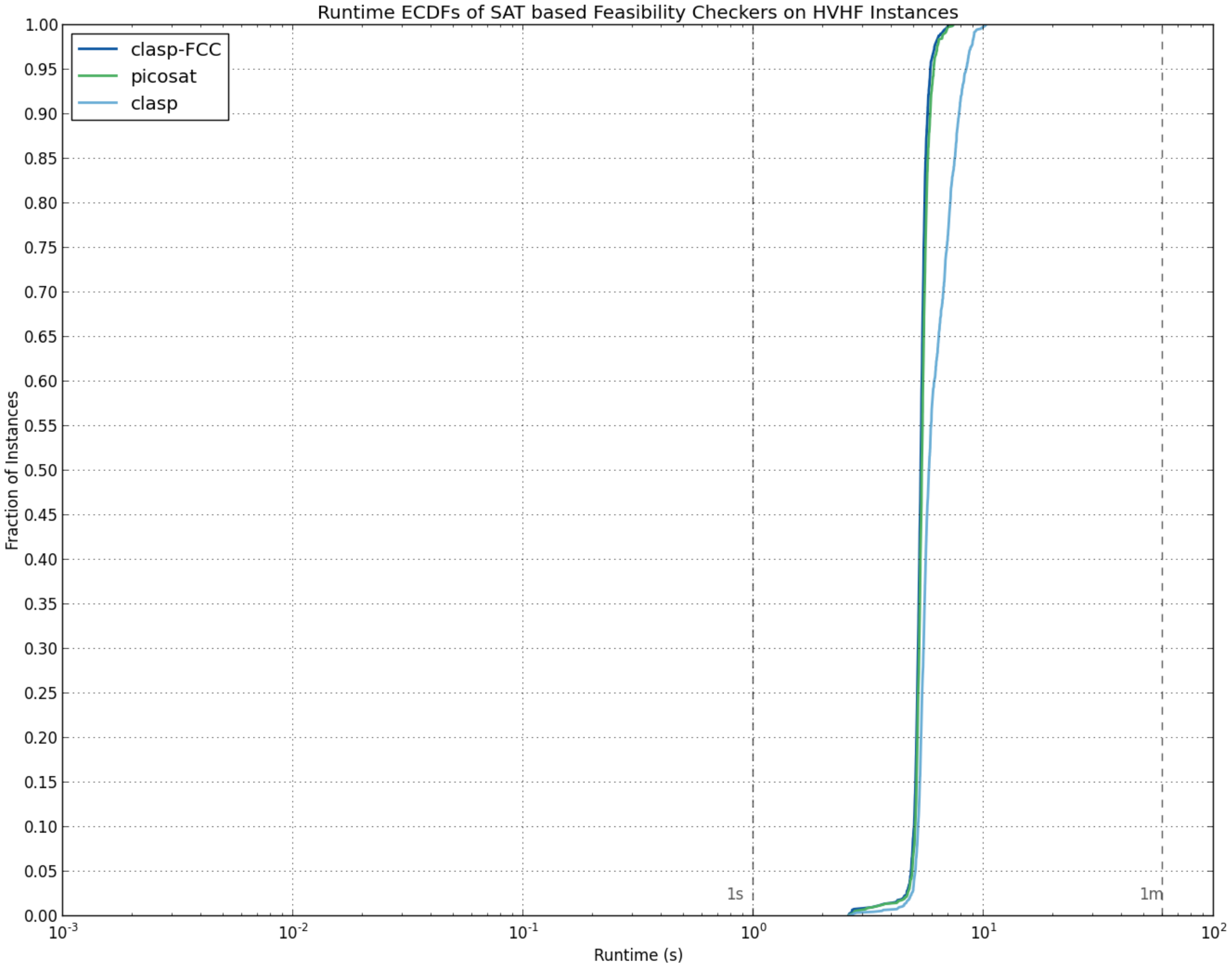


SATFC: VHF PERFORMANCE

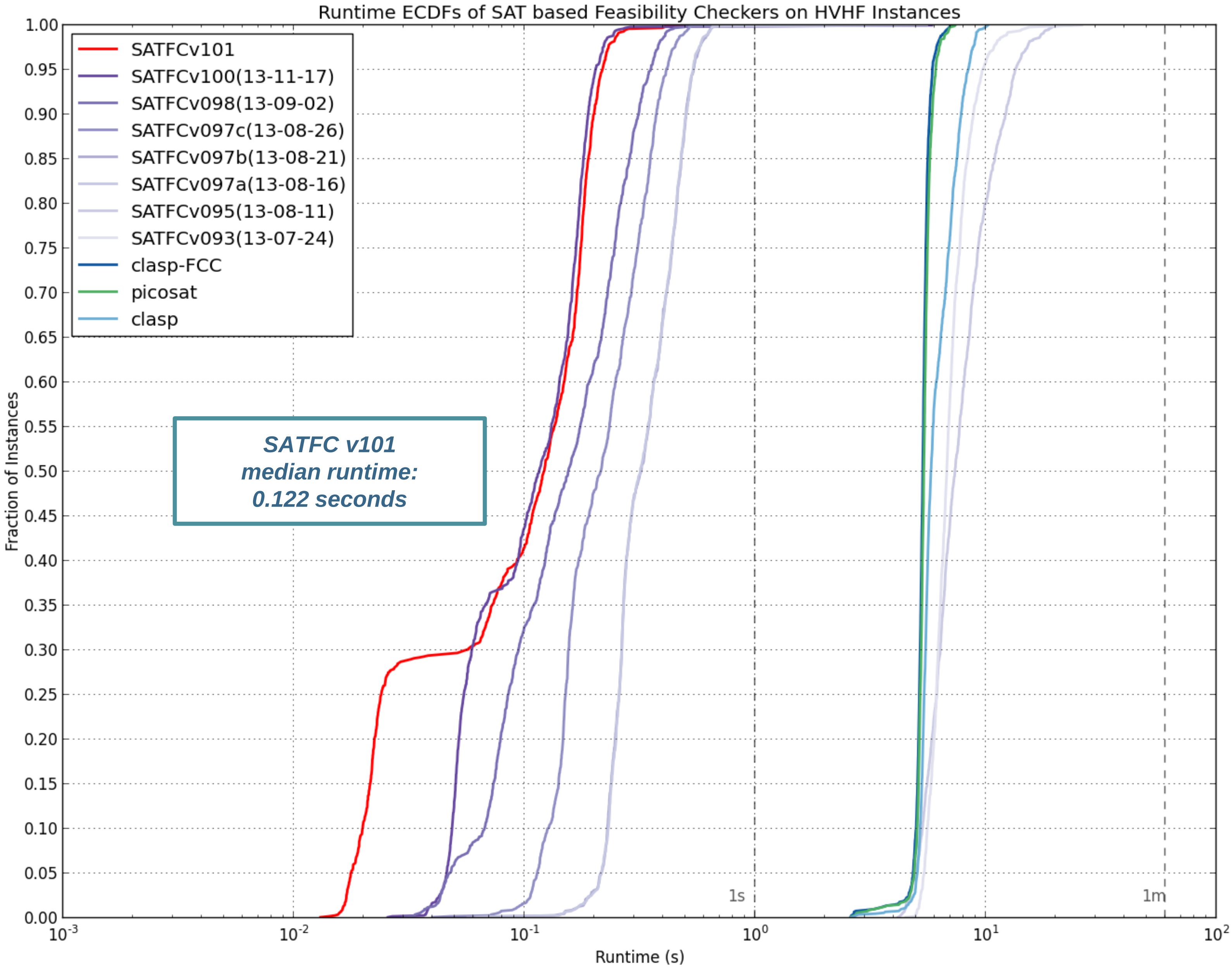
VHF: Comparing off-the-shelf SAT solvers



VHF: Adding our specially-configured version of clasp

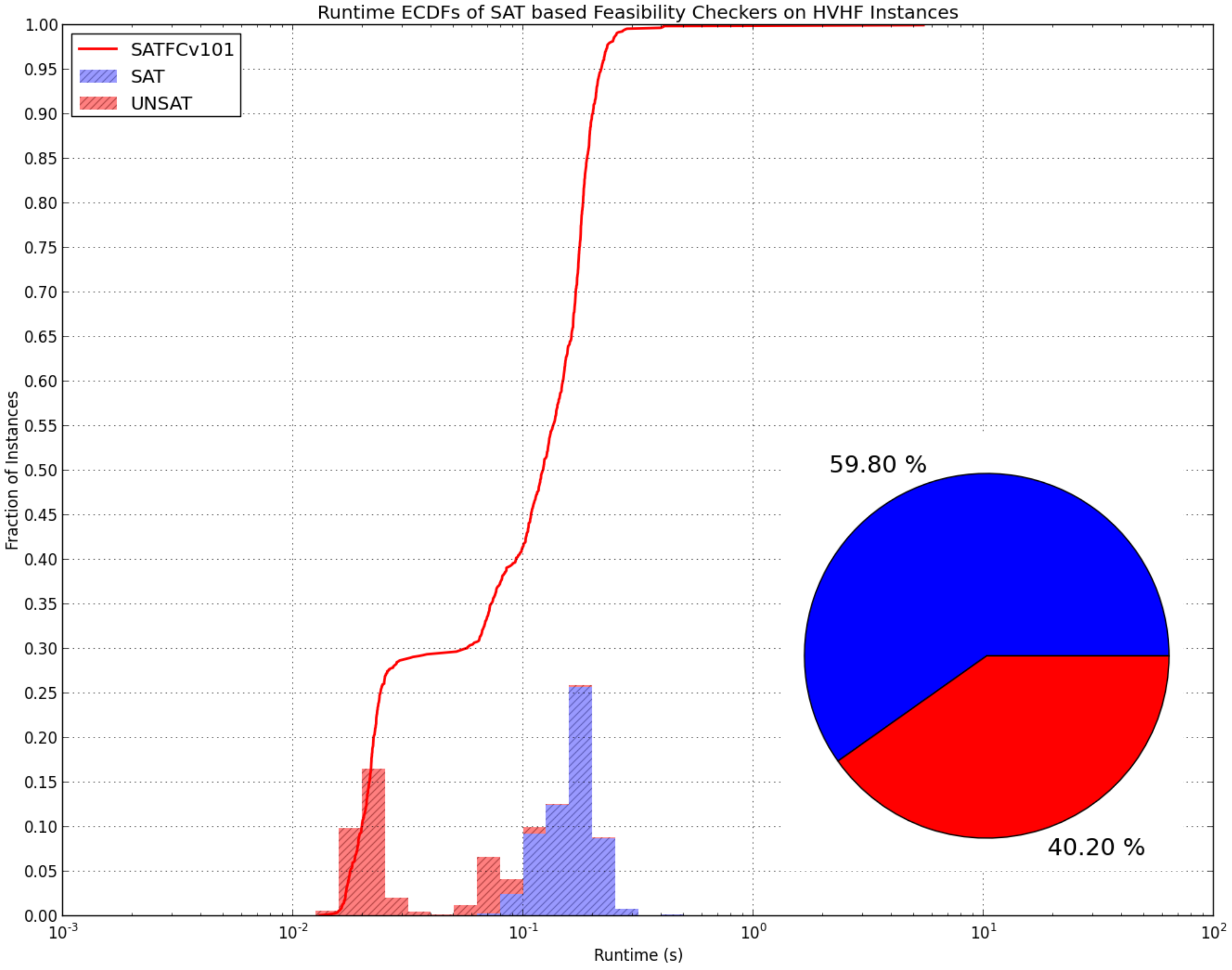


VHF: SATF performance improvements over time (July 24 - present)



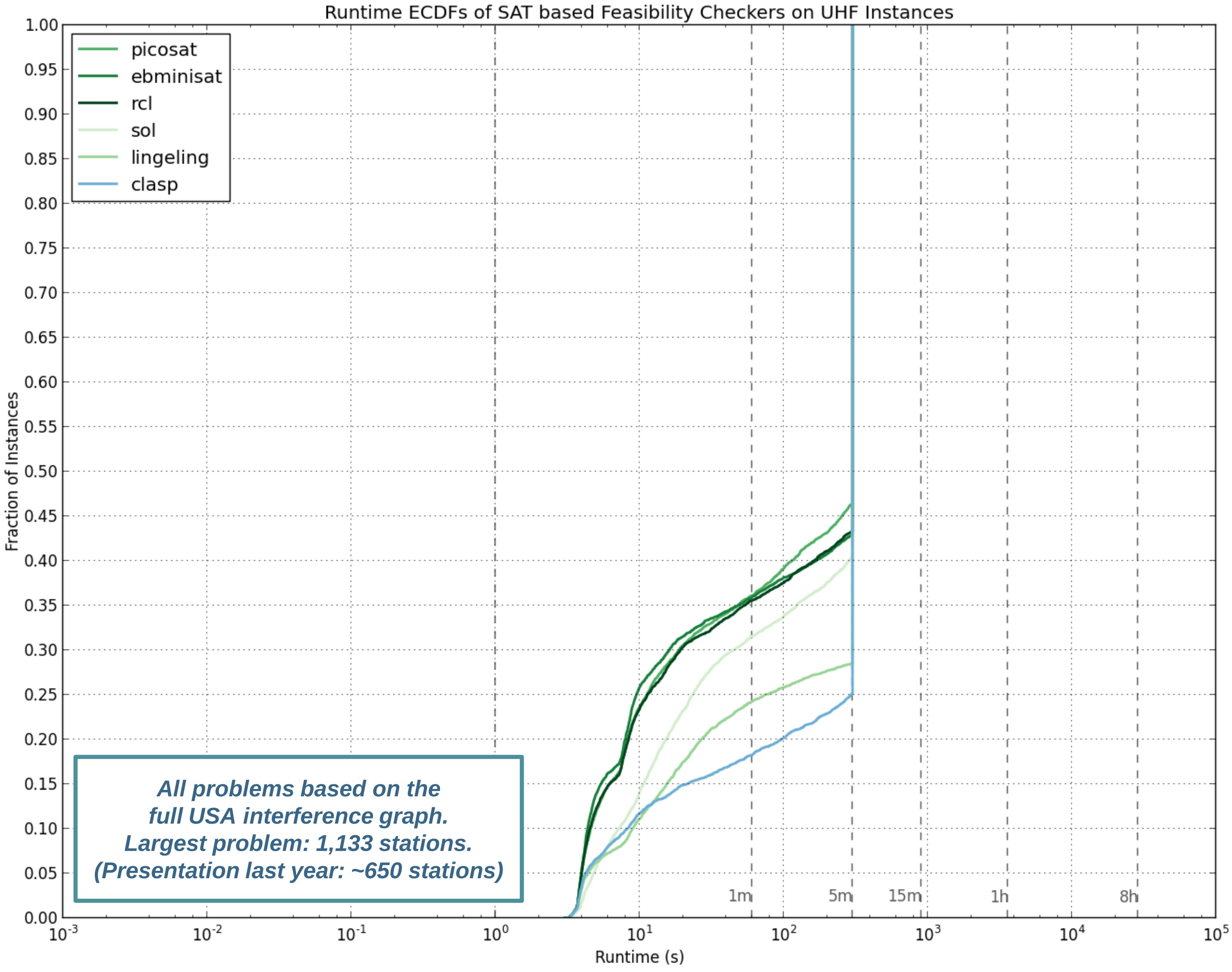
SATFC v101
median runtime:
0.122 seconds

VHF: SATFC performance and SAT/UNSAT breakdown



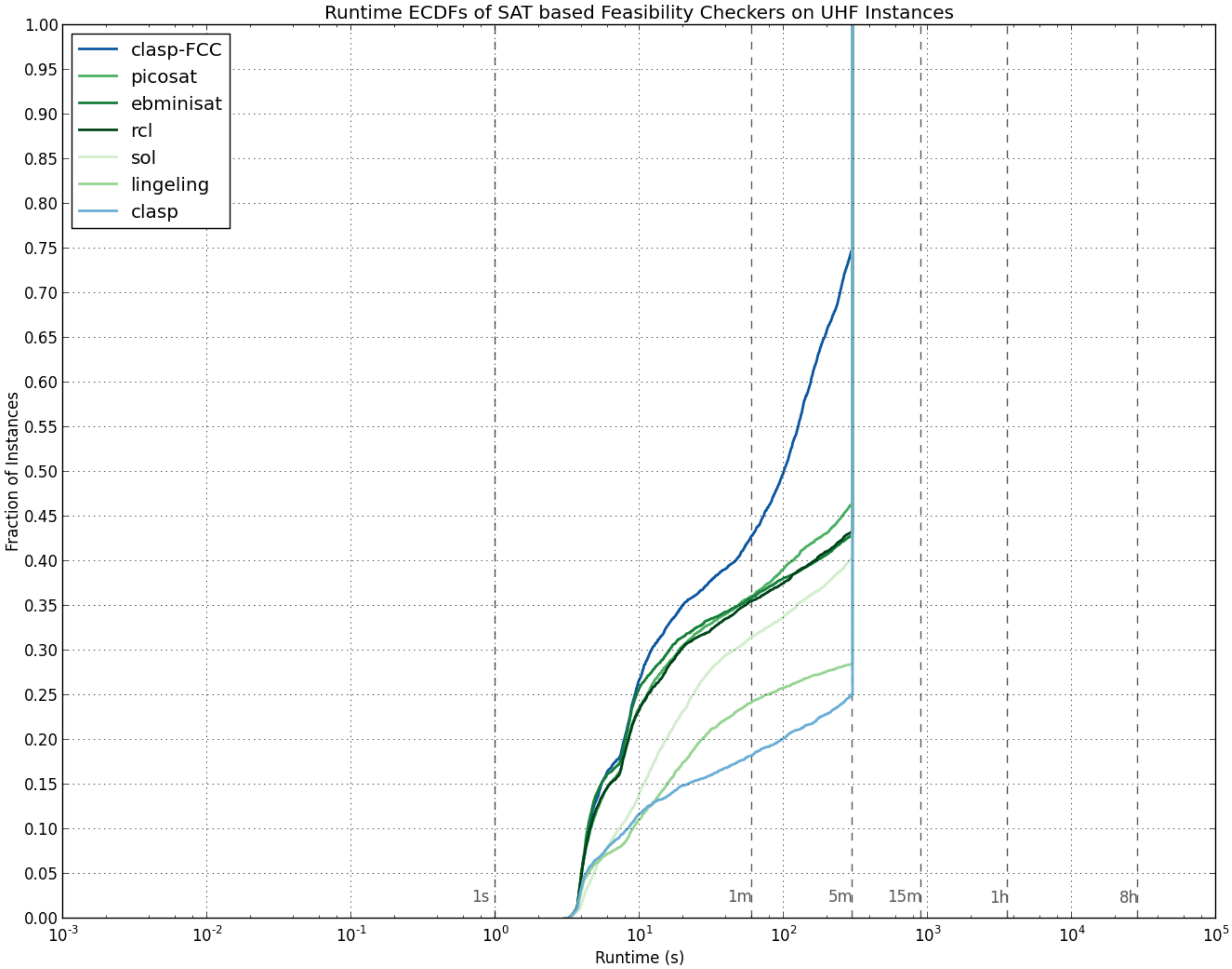
SATFC: UHF PERFORMANCE

UHF: Comparing off-the-shelf SAT solvers (5 min cutoff)

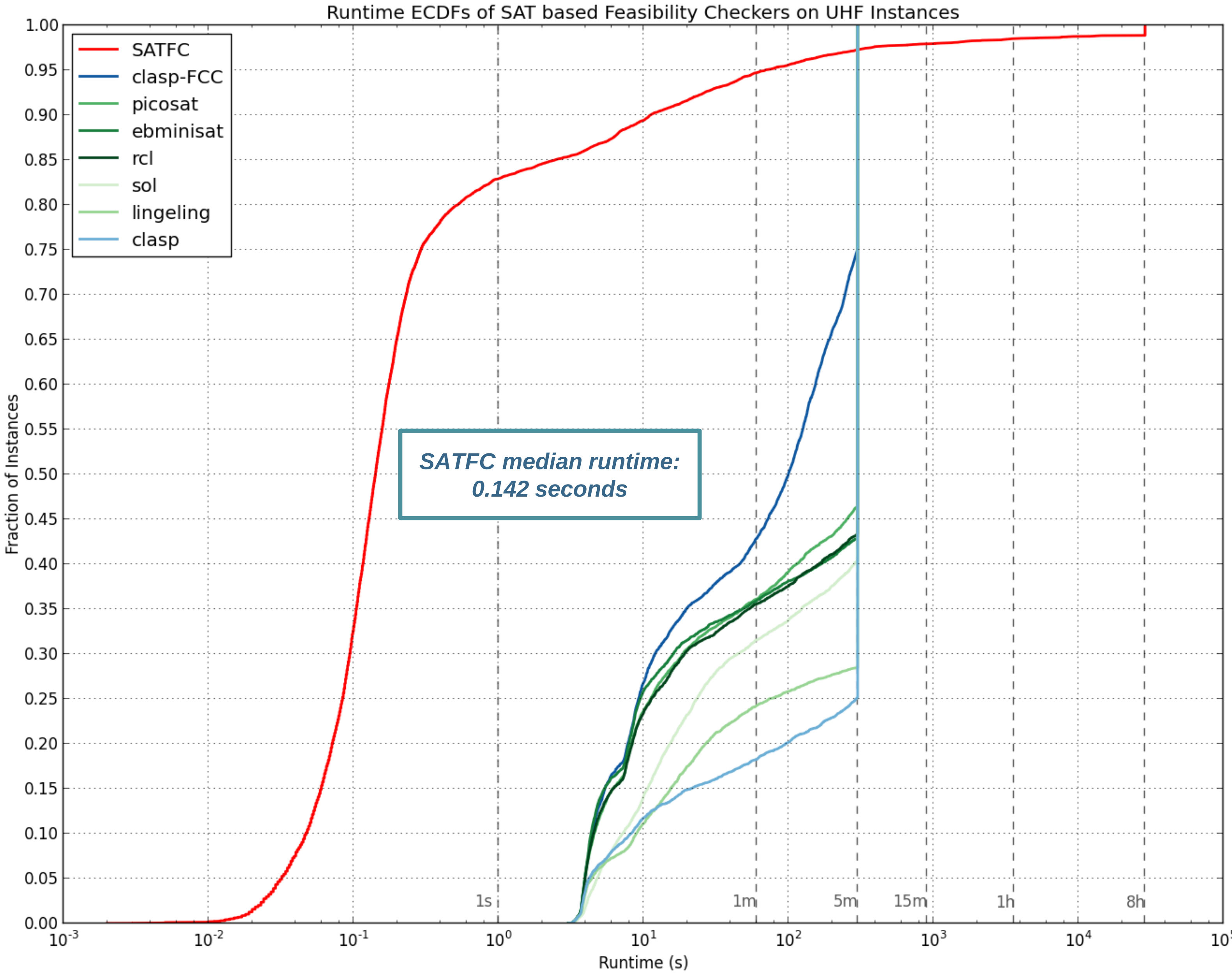


All problems based on the full USA interference graph. Largest problem: 1,133 stations. (Presentation last year: ~650 stations)

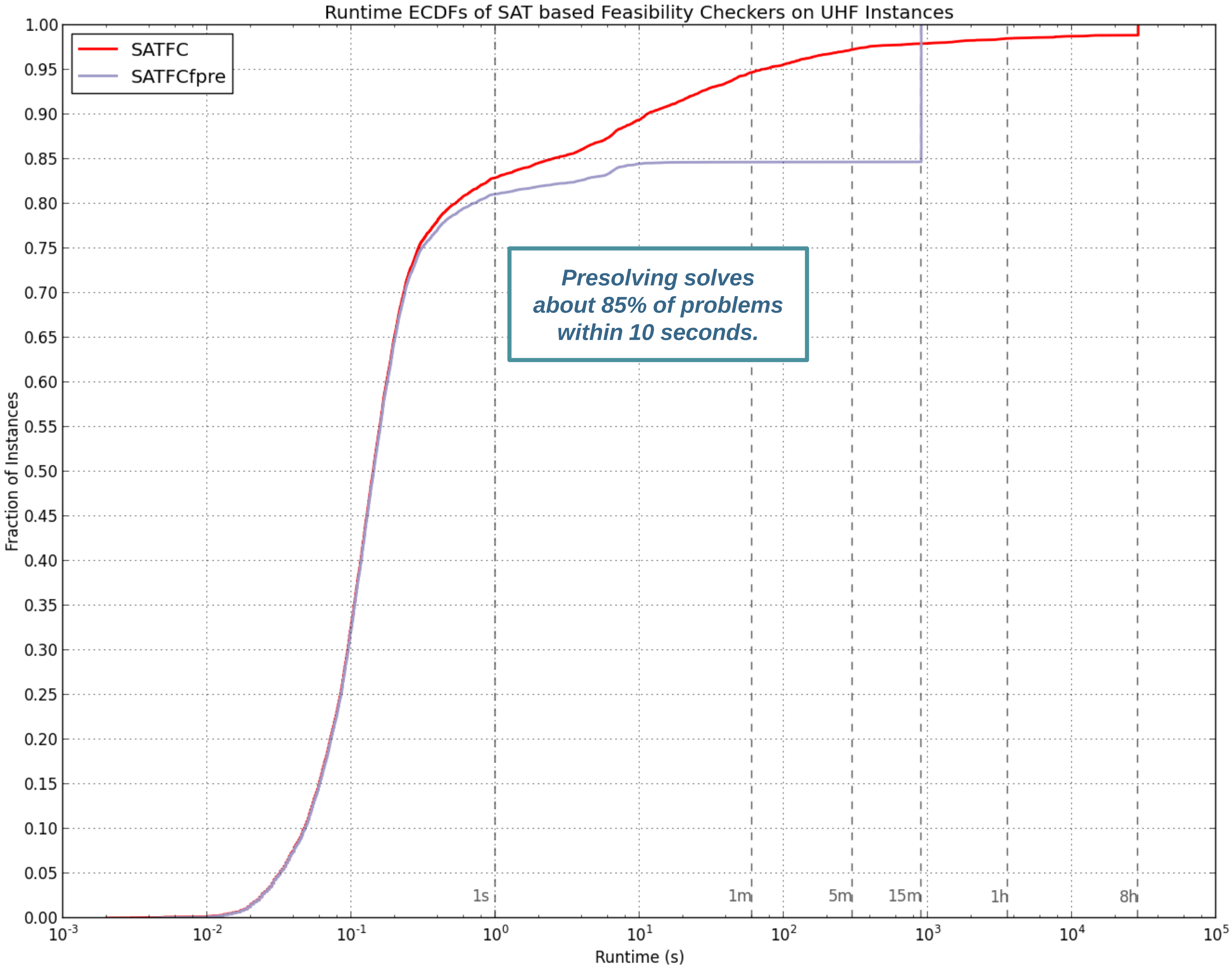
UHF: Adding our specially-configured version of clasp



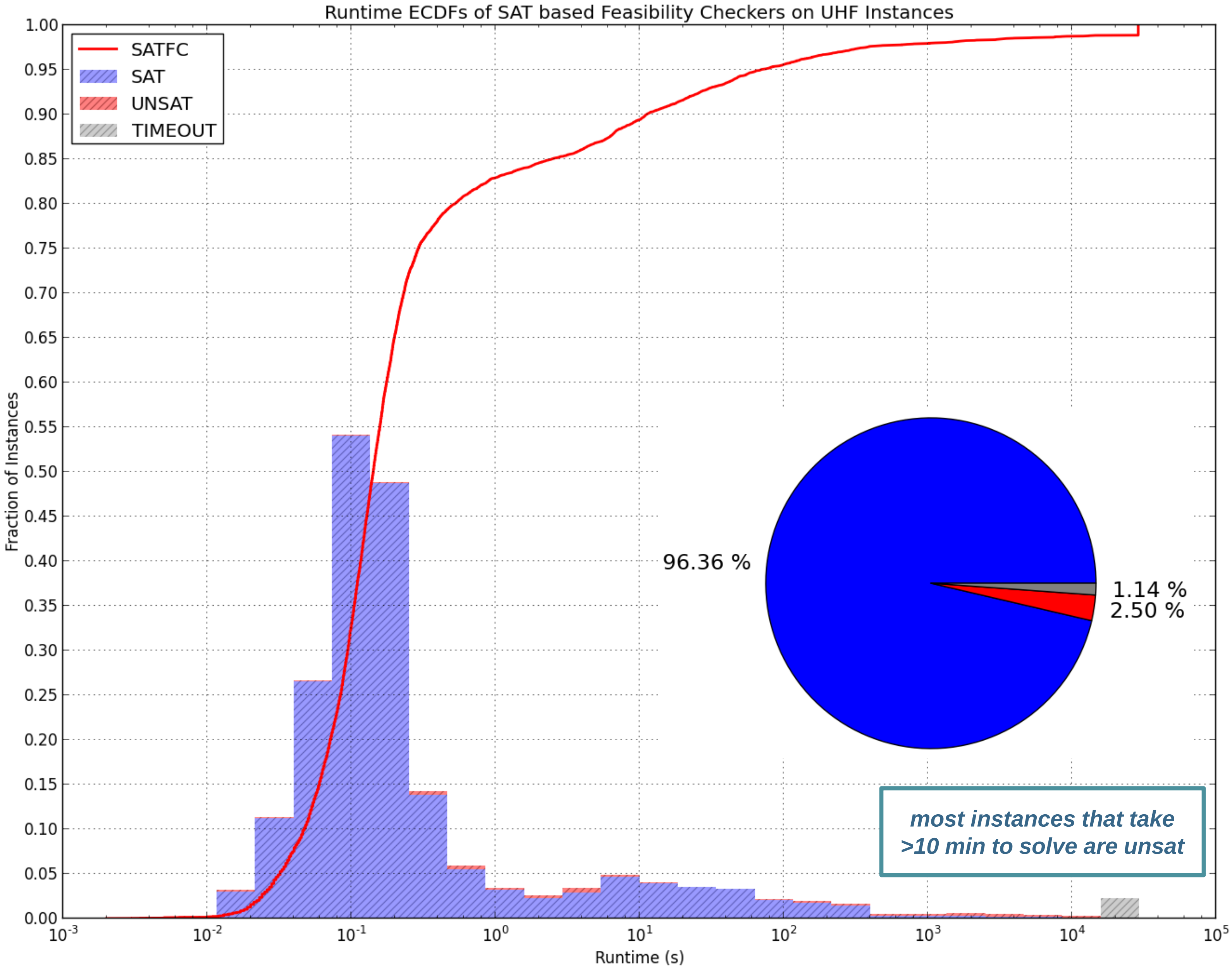
UHF: Adding the latest version of SATFC *(8h cutoff)*



UHF: The effect of presolving on SATFC performance



UHF: SATFC performance and SAT/UNSAT breakdown



most instances that take >10 min to solve are unsat

QUESTIONS?

Integer Optimization Solvers

- Also known as Integer Linear Programming
- Find an optimal solution to a linear objective function subject to a set of linear constraints
- Some or all decision variables restricted to integer values

Integer Optimization Solvers

- For TV station feasibility checking we are seeking only a feasible solution
- Objective function can be the zero vector
- Define linear constraints to enforce interference restrictions
- Decision variables define allowable channel assignments

Integer Optimization Formulation

Definition of indices:

S = the set of all stations to be assigned

C_s = the domain set for station $s \epsilon S$, i.e. the set of allowable channels in the repacking band

Definition of variables:

$$x_{s,c} = \begin{cases} 1 \text{ if station } s \epsilon S \text{ is assigned to channel } c \epsilon C_s \\ 0 \text{ otherwise} \end{cases}$$

Integer Optimization Formulation

The constraints

For each station one of its allowable channels must be assigned to the station:

$\sum_{c \in C_s} x_{s,c} = 1, \forall s \in S$

For every co-channel pairwise restriction at most one of the two stations can be assigned to that channel:

$x_{l,c} + x_{m,c} \leq 1$

For every adjacent pairwise restriction at most one of the two stations can be assigned to the that channel:

$x_{l,c} + x_{m,c+1} \leq 1$

Integer Optimization Formulation

- The feasibility checker problem has a very special structure
- Pairwise constraints can be combined into much stronger constraints known as "clique constraints"
- A clique is a set of variables that has the property that only one variable in this set can be set to 1 (or true)
- Significantly reduces the number of constraints on the problem