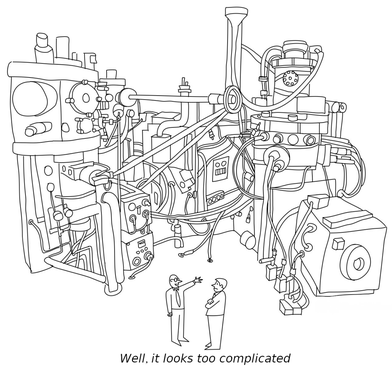
Scalable Anytime Algorithms for Learning Formulas in Linear Temporal Logic

Ritam Raha, Rajarshi Roy, Nathanaël Fijalkow & Daniel Neider

ETAPS European Joint Conferences on Theory & Practice of Software
TACAS 2022

LaBRI

1. Motivation



Well, it looks too complicated

Goal: Learn Simple (human interpretable) models by observing complex systems

2. Linear Temporal Logic

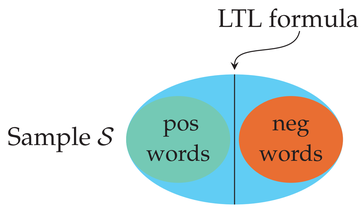
- Interpreted over **finite words**
- Used to express temporal properties using *Finally* (**F**), *Globally* (**G**), *Next* (**X**), etc. operator.

Syntax:

$$\varphi ::= p \in \Sigma \mid \neg p \mid \varphi \vee \varphi \mid \varphi \wedge \varphi \mid \mathbf{X}\varphi \mid \mathbf{F}\varphi \mid \mathbf{G}\varphi$$

- LTL is close to natural language
- Until (**U**) is not our fragment.

3. The Learning Problem



- **Input:** A set of positive words P & negative words N
- **Question:** Find a *minimal separating* LTL formula φ, such that, all $w \in P$ satisfies φ and all $w \in N$ do not.

4. State of the Art

Existing approaches use:

- SAT-Solvers (Neider & Gavran, FMCAD'18)
- SyGuS solvers (Arif et al., FMCAD'20)

5. Overview of the Algorithm

- Use **Dynamic Programming** to extract LTL patterns of increasing size
- Use **Greedy approximation** to generate their Boolean combinations (**Boolean Set Cover**)

5.1. Extracting LTL patterns

Directed LTL (dLTL)

LTL patterns with the following grammar:

$$\varphi: \; = \mathbf{X}^n p \mid \mathbf{F}\mathbf{X}^n p \mid \mathbf{X}^n(p \wedge \varphi) \mid \mathbf{F}\mathbf{X}^n(p \wedge \varphi)$$

Example: $\mathbf{F}(p \wedge \mathbf{X}q)$ is in dLTL, but $(\mathbf{F}p \wedge \mathbf{X}q)$ is not.

Extracting dLTL formulas

Sample S

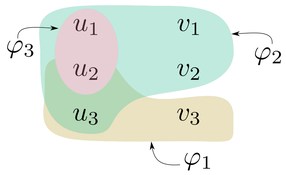
Positive Words	Negative Words
pqqqpp ✓	qqqq ✘
qqqppp ✓	ppqp ✘

Idea: Find *separating patterns* **with intervals**

Candidate: 1q>0p

Formula: X(q ∧ Fp)

5.2. Boolean Set Cover



Problem: Find the minimal separating *boolean combination of formulas*
Algorithm: Extension of classical Set Cover: Greedy approximation
Solution: $(\varphi_1 \wedge \varphi_2) \vee \varphi_3$

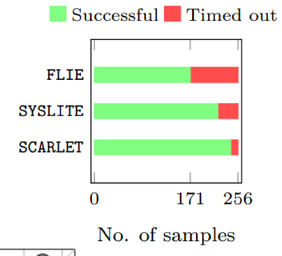
Theoretical Guarentees

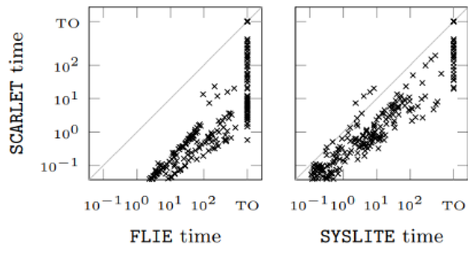
The boolean combination of dLTL formulas is as expressive as LTL(**F**, **X**, ∧, ∨)

Dual: $\neg\mathbf{F}\neg\varphi = \mathbf{G}\varphi$: swap positive and negative words- can capture **G**!

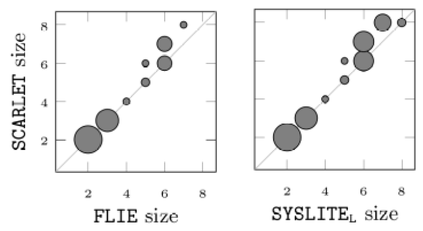
6. Experimental Evaluation

- `SCARLET`: python based tool; compared it against two tools `FLIE` (SAT-based) and `SYSLITE` (SyGuS based)
- Benchmarks synthetically generated from existing formulas without **U** operator.





Runtime comparison



Size comparison

7. Conclusion

- An approximation algorithm for finding concise separating LTL formulas
- Empirically established that such an algorithm infers concise formulas in reasonable time

Contact Information

Ritam Raha
Email: ritam.raha@uantwerpen.be

Rajarshi Roy
Email: rajarshi@mpi-sws.org