FMitF: Track I: Principles for Modular Probabilistic Programming and Inference

Northeastern University

Challenge:

It is extremely difficult to reason about **large-scale probabilistic systems** due to the inherent state-space explosion of probabilistic behavior

→ **Consequence**: difficult to verify probabilistic systems (for instance, in machine learning)

Solution: New approaches for modularly decomposing large probabilistic systems into smaller ones.

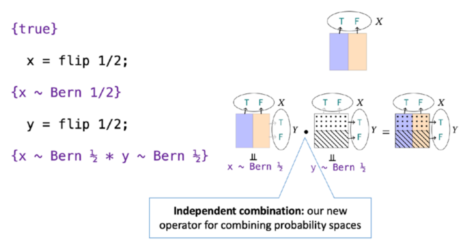
- **Lilac**: a modal separation logic for conditional probability. Decompose probability with separation logic
- **MultiPPL**: a multi-language approach to proababilistic programming language design. Decompose probabilistic reasoning along language boundaries.

Award #2220408

PI: Steven Holtzen s.holtzen@northeastern.edu

Co-PI: Amal Ahmed amal@ccs.neu.edu

Lilac: A Probabilistic Separation Logic for Conditional Probability



MultiPPL: Multi-Language Probabilistic Programming

```
let x be flip 0.20 in
(|let Y be flip 0.25 in
 observe (|x|)_E ∨ Y in
 ret Y|)_S
```

Scientific Impact:

- New approaches and foundations to probabilistic separation logic that enable scalable modular verification of probabilistic systems
- More scalable and broadly applicable probabilistic verification for machine learning and randomized algorithms
- More scalable and automated probabilistic programming languages

Broader Impact and Broader Participation:

- *Broad impact*: More usable machine learning and probabilistic reasoning for broad audiences
- *Deployment*: tools deployed at Sandia National Laboratories, collaboration on papers.
- *Educational impact*: new course on probabilistic programming at the Oregon Programming Languages Summer School 2024, presentations at the Programming Languages Mentorship Workshop, new course on probabilistic programming languages at Northeastern University