A Generalization of SAT and #SAT for Probabilistic Inference and Verification

Carnegie Mellon
Erik Zawadzki
Geoffrey Gordon
André Platzer

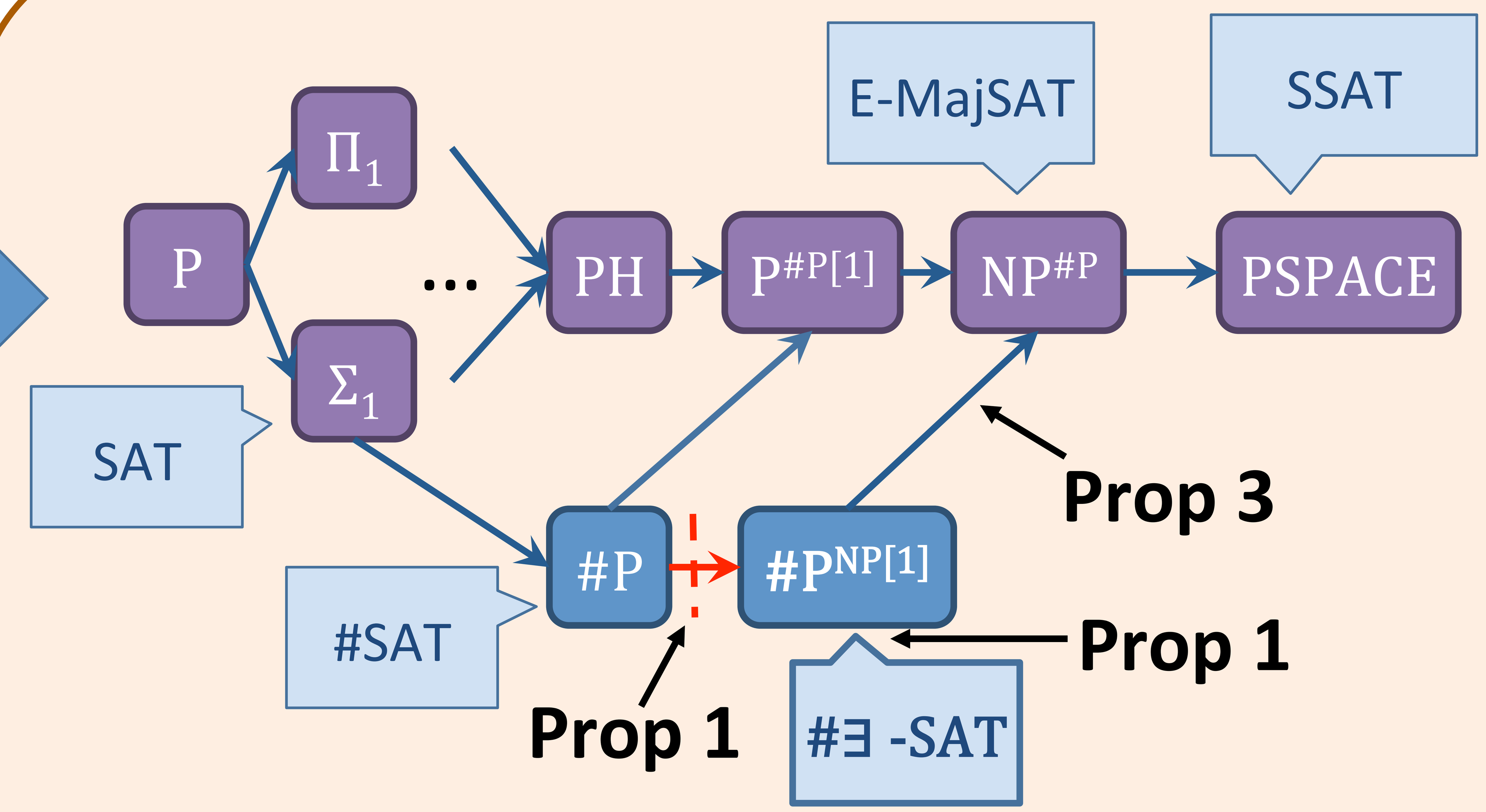
Alternations and #∃-SAT

$\Phi:\{T,F\}^n \rightarrow \{T,F\}$

$Q \in \{\exists, \forall, \Sigma, \ldots\}^n$ — $Q_x \Phi(x)$ — $x : \{T,F\}^n$

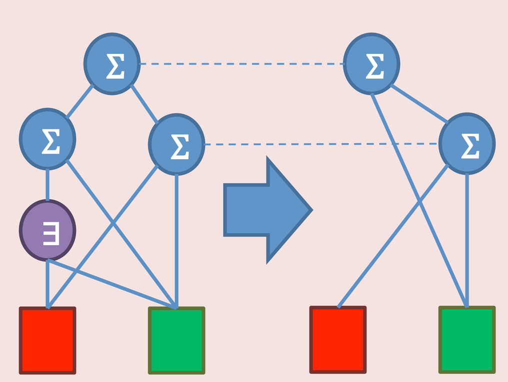
- $Q=\exists^n$ can encode decision making (SAT)
- $Q=\Sigma^n$ can encode probabilities (#SAT)
- **With $Q = \Sigma^m \exists^{n-m}$ we can express #∃-SAT, a new type of problem for robust verification**

Theory



Algorithms

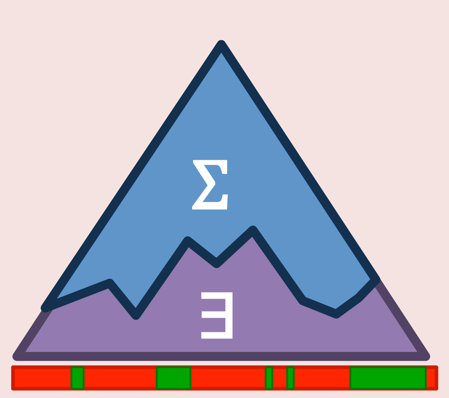
BDD



1. Build a BDD for the formula Φ in stratified variable order
2. Eliminate ∃-nodes and simplify
3. Count remaining nodes

Only useful if base BDD can be constructed cheaply

DPLL



- Two-level DPLL search (∃ and Σ)
- With or without component

Heavy-weight component caching does not seem useful

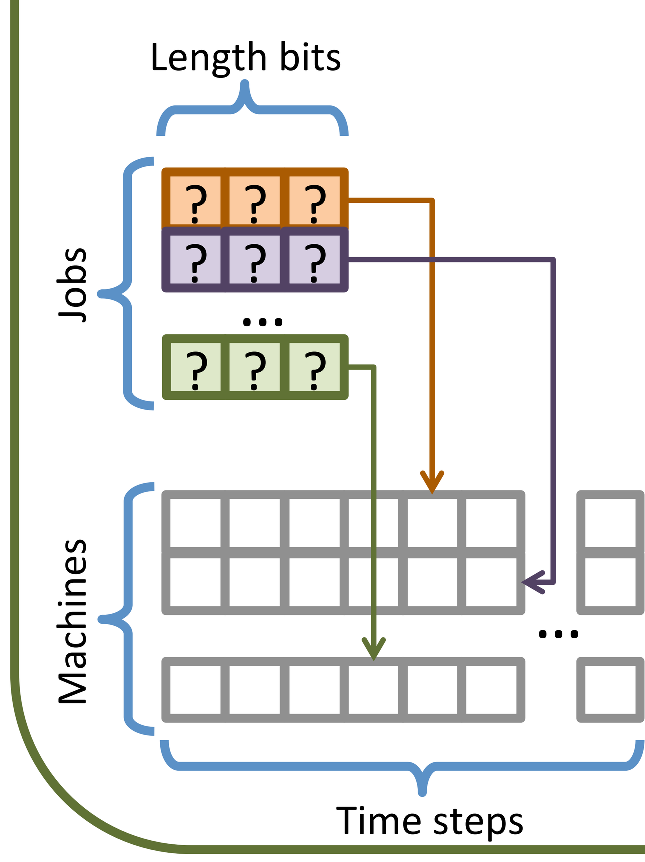
POPS

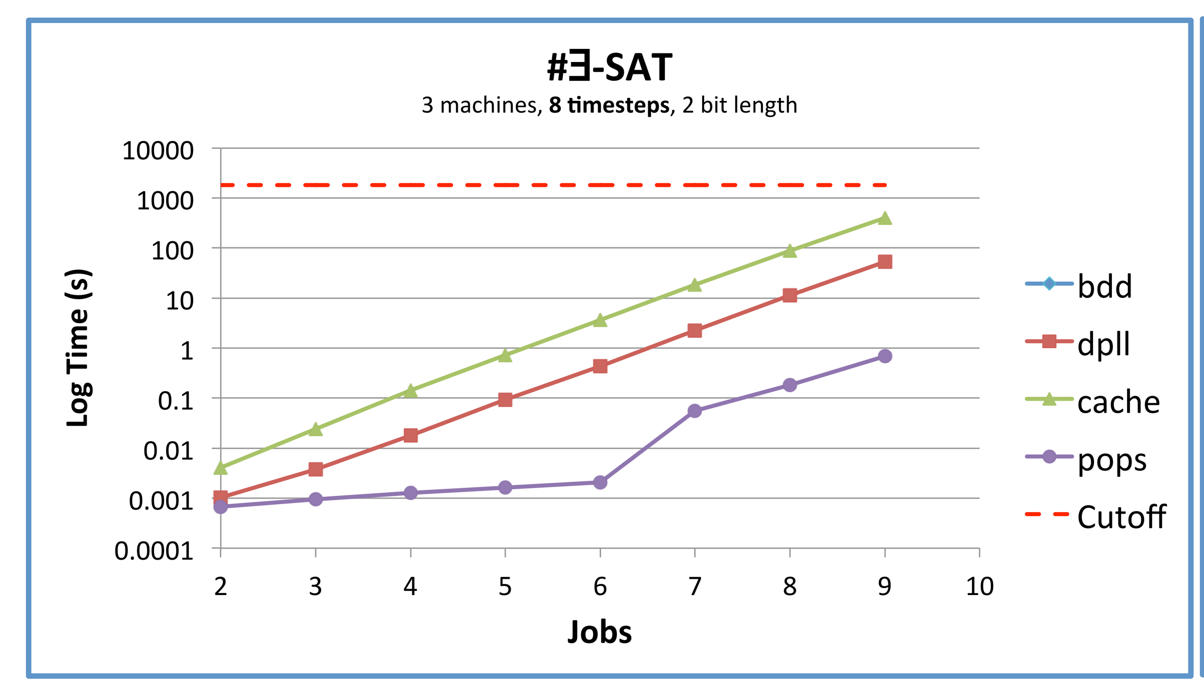
	x	¬x
Lucky	T	T
True	T	F
False	F	T
Unlucky	F	F

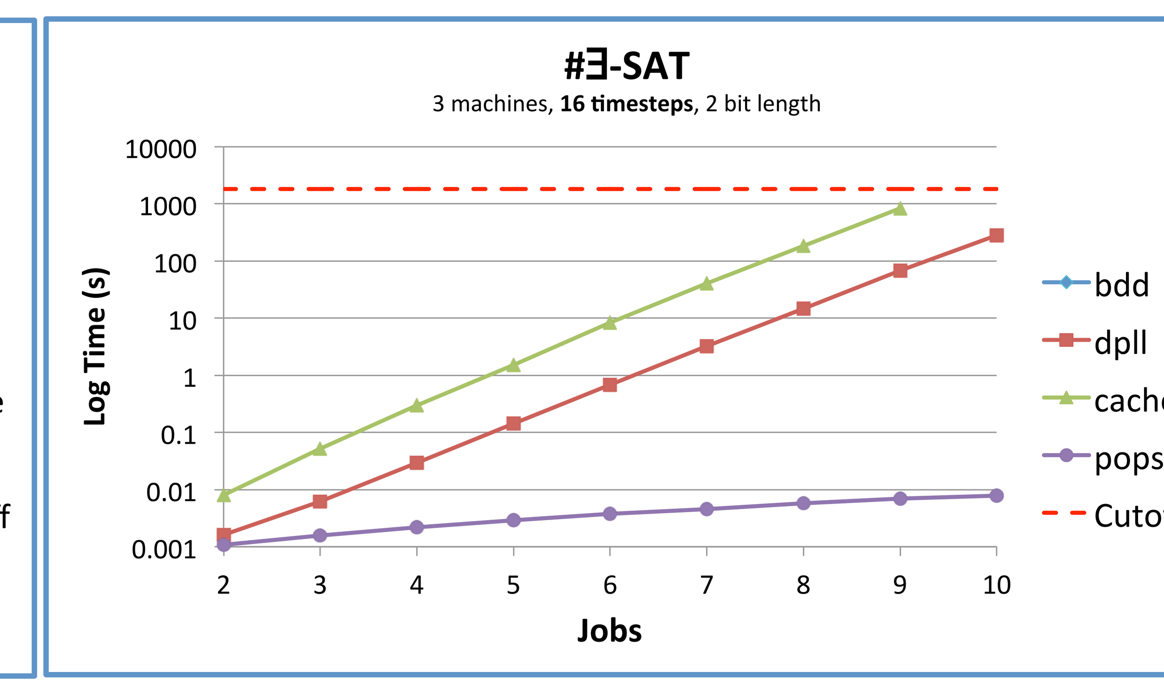
- Prune with pessimistic and optimistic SAT subproblems
- Uses a 4-valued logic: lucky, true, false, unlucky

Can exploit a type of structure other solvers cannot

Sample experiment: Job shop scheduling







Experiments

Example