UNIVERSITY OF WASHINGTON

Automated Basis Functions for Goal-Directed MDPs

Mausam

University of Washington, Seattle

Joint work with Andrey Kolobov and Dan Weld

Setting

- I know the model -- factored MDP
  - PPDDL (logical specified domains)
- I want to achieve a goal
  - Goal may be hard to reach
- Problem is too large…
  - Can't enumerate all states
  - VI (or other flat algorithms) impractical

Exploit structure in PPDDL effectively
declaratively exploses causal structure

Previous Work

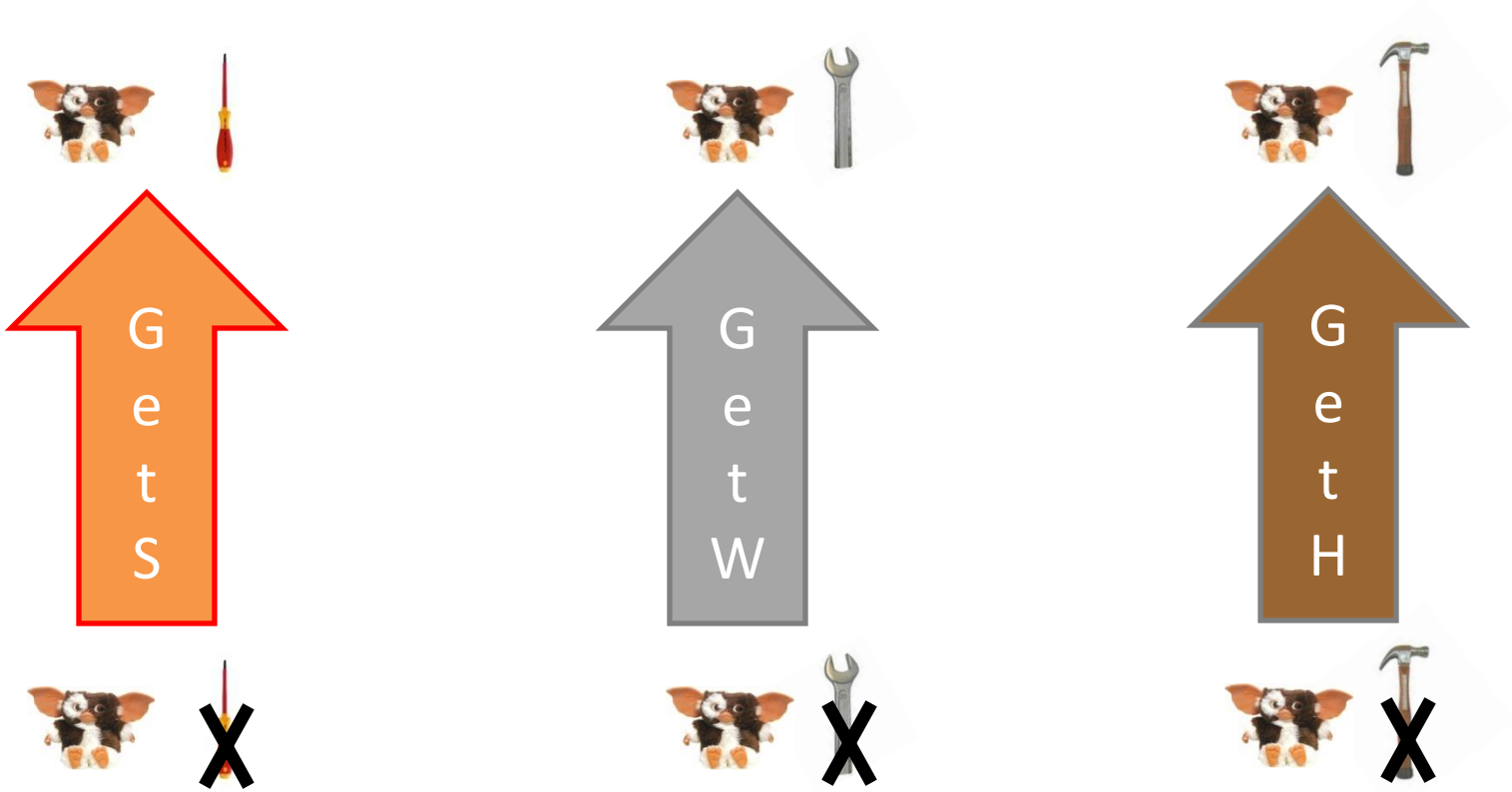
- Determinization
  - Determinize the MDP
  - Classical planners ***fast***
  - E.g., FF-Replan
  - Cons: may be troubled by
    - Complex contingencies
    - Probabilities
- Function Approximation
  - Dimensionality reduction
  - Represent state values with basis functions
    - E.g., $V^*(s) \approx \sum_i w_i b_i(s)$
  - Cons:
    - Need a human to get b_i

Marry these paradigms to extract problem-specific structure in a fast, problem-independent way.

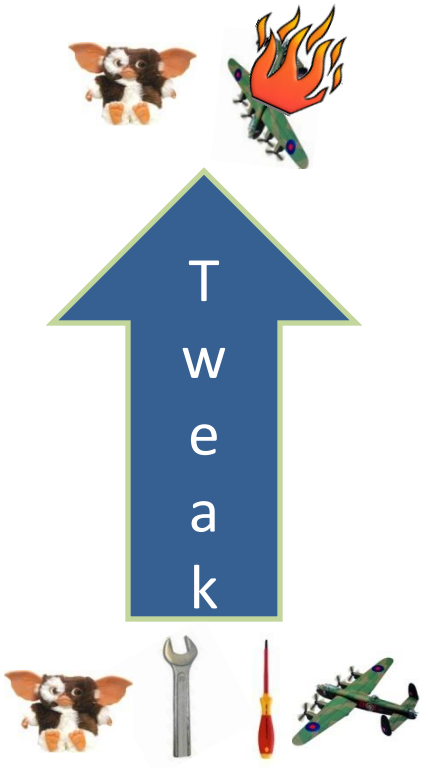
Outline

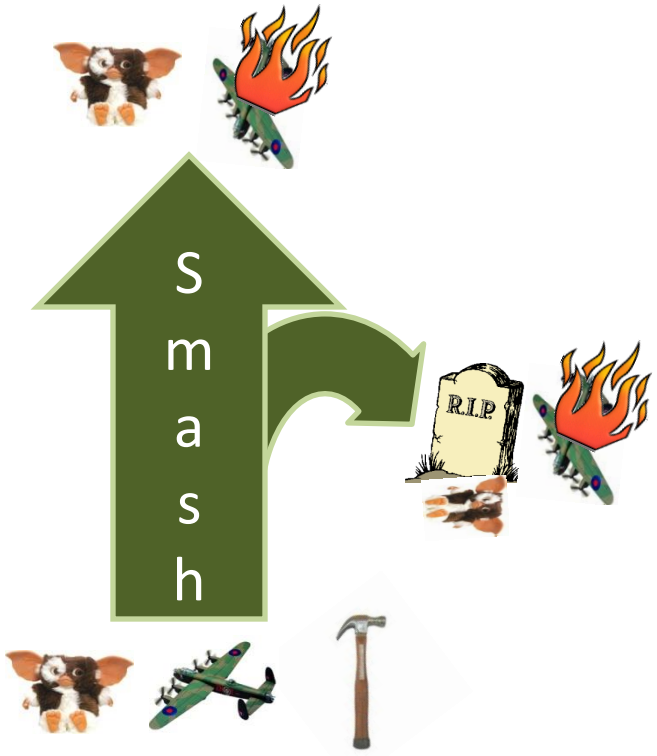
- Motivation
- Running Example
- ReTrASE: Basis Function Generation
- SixthSense: Dead-end Generalization
- Conclusions

Example Domain



Example Domain (cont'd)

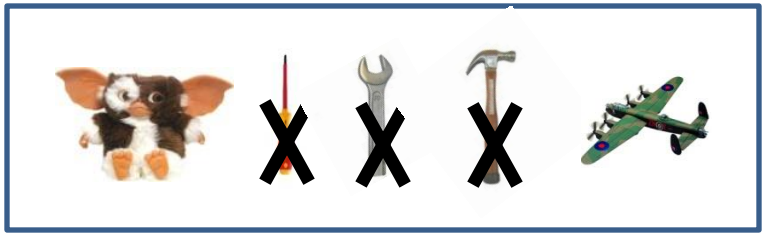




Markov Decision Process (MDP)

- S: A set of states
- A: A set of actions
- Pr(s'|s,a): transition model
- C(s,a,s'): action cost
- s_0: start state
- G: set of goals

GetW, GetH, GetS, Tweak, Smash



Outline

- Motivation
- Running Example
- ReTrASE: Basis Function Generation
- SixthSense: Dead-end Generalization
- Conclusions

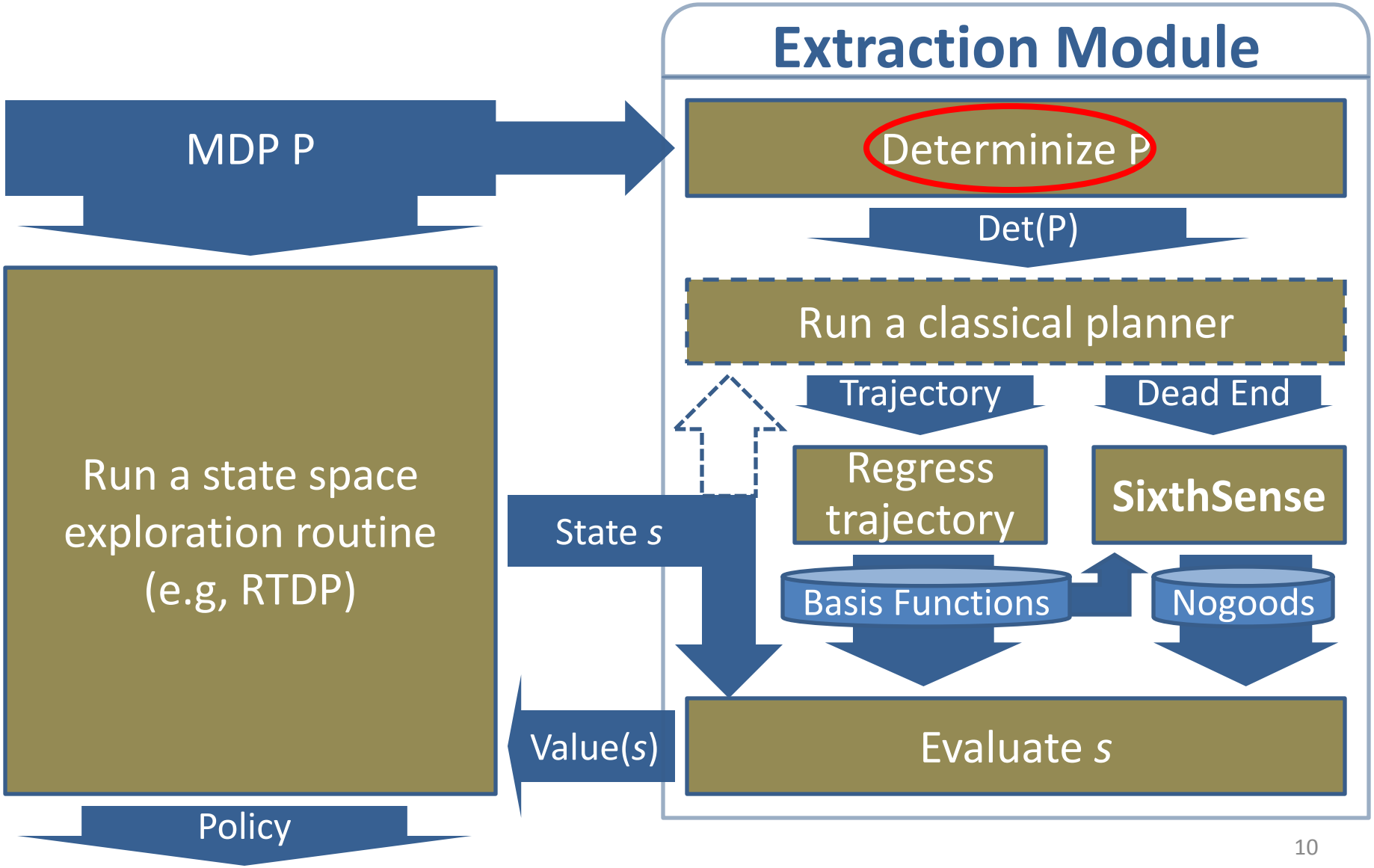
Contributions

ReTrASE — a *scalable* approximate MDP solver

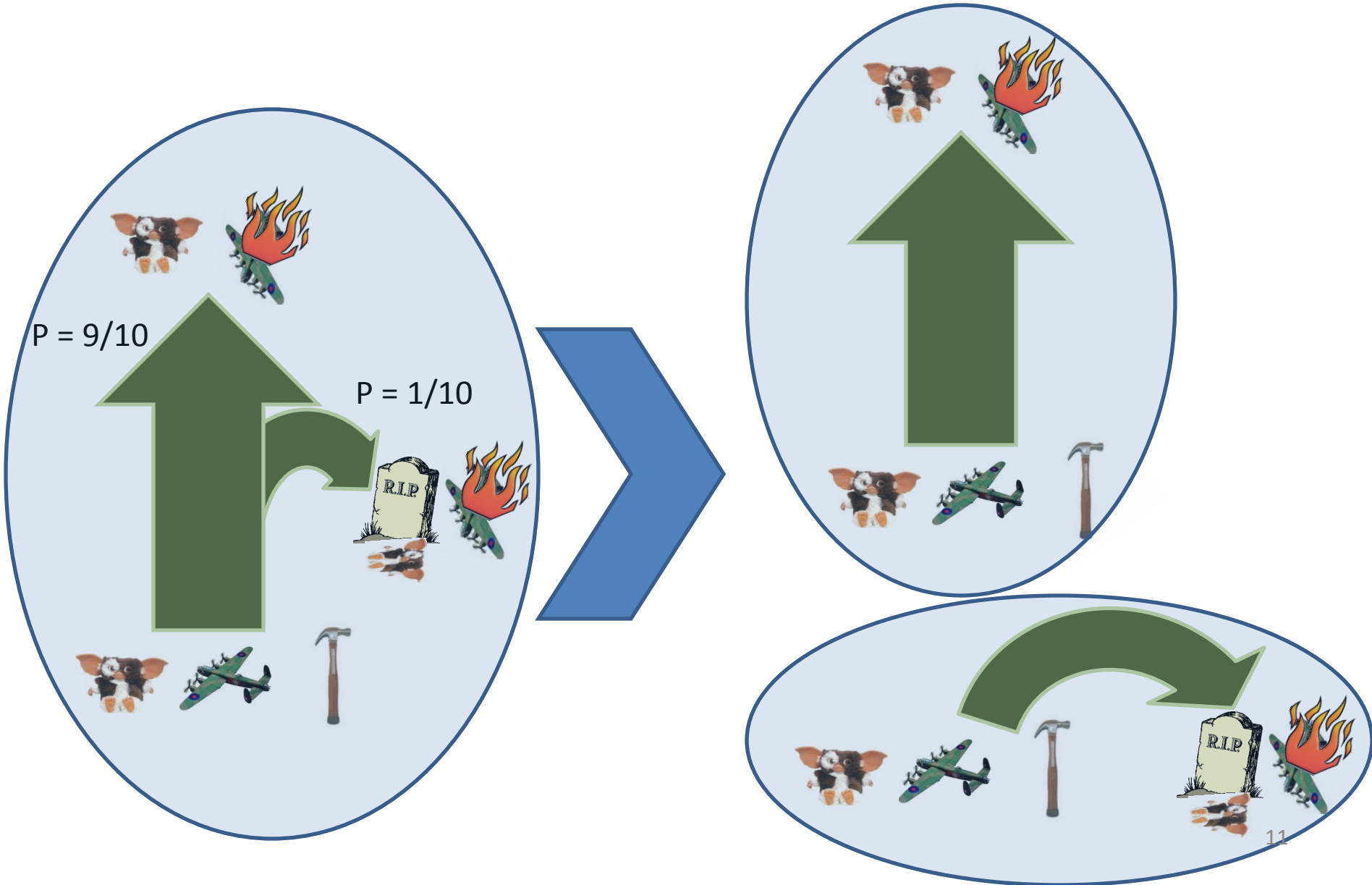
- Combines function approximation with classical planning
- Uses classical planner to automatically generate basis functions
- Fast, memory-efficient, high-quality policies

The Big Picture: **ReTrASE**

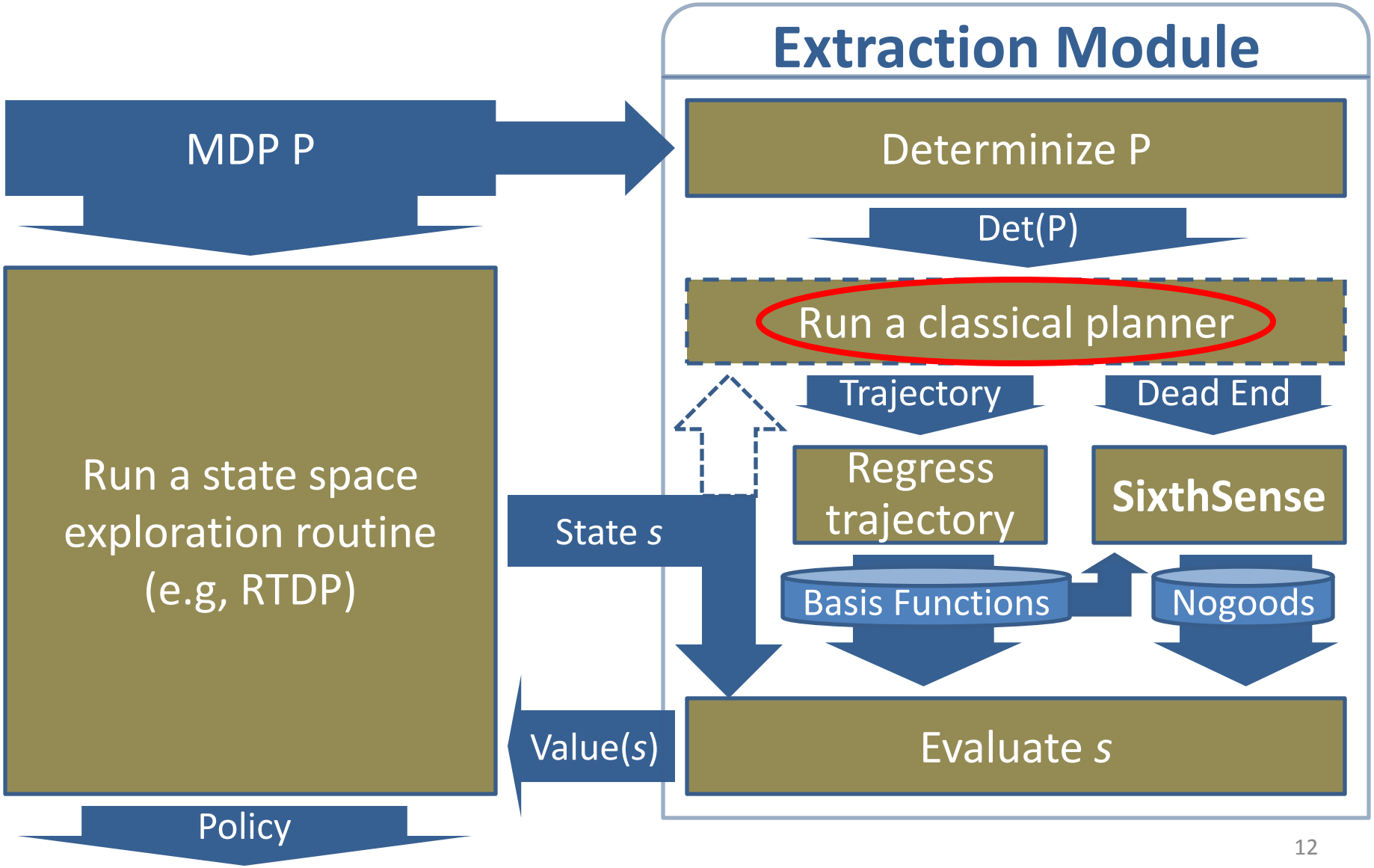
Kolobov, Mausam, Weld, AIJ'12



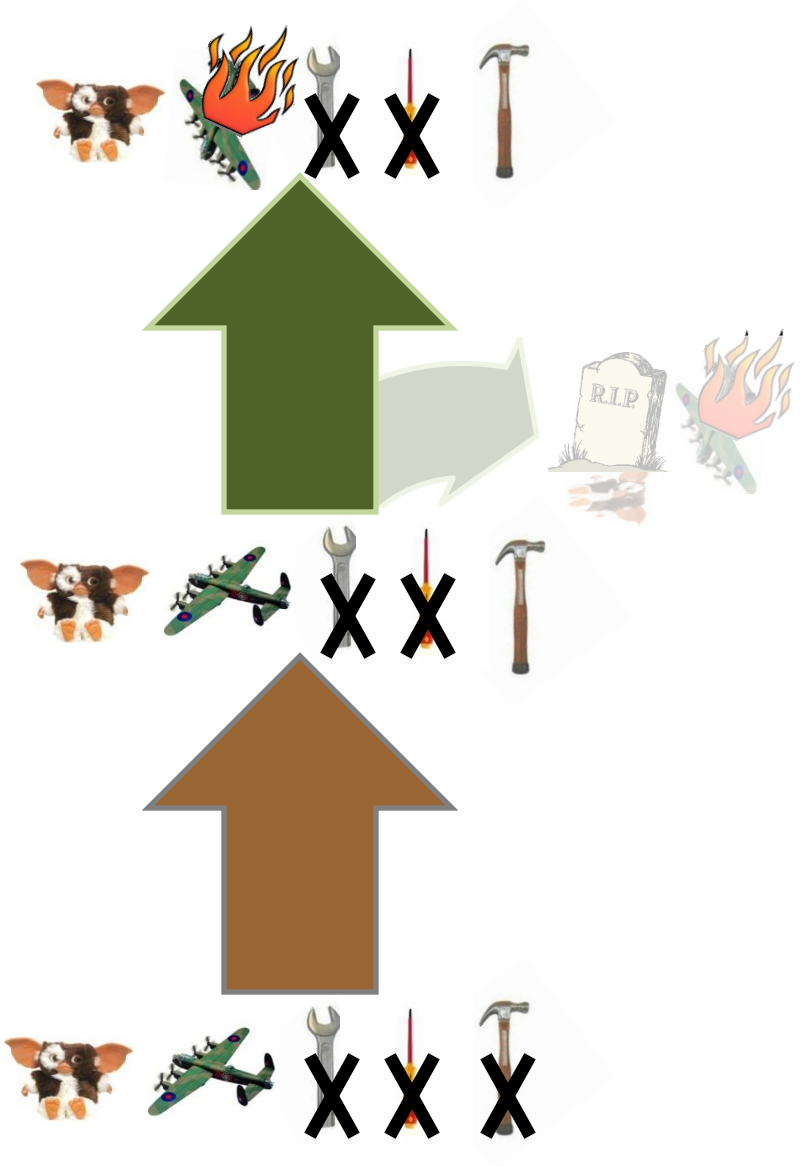
Determinizing the Domain



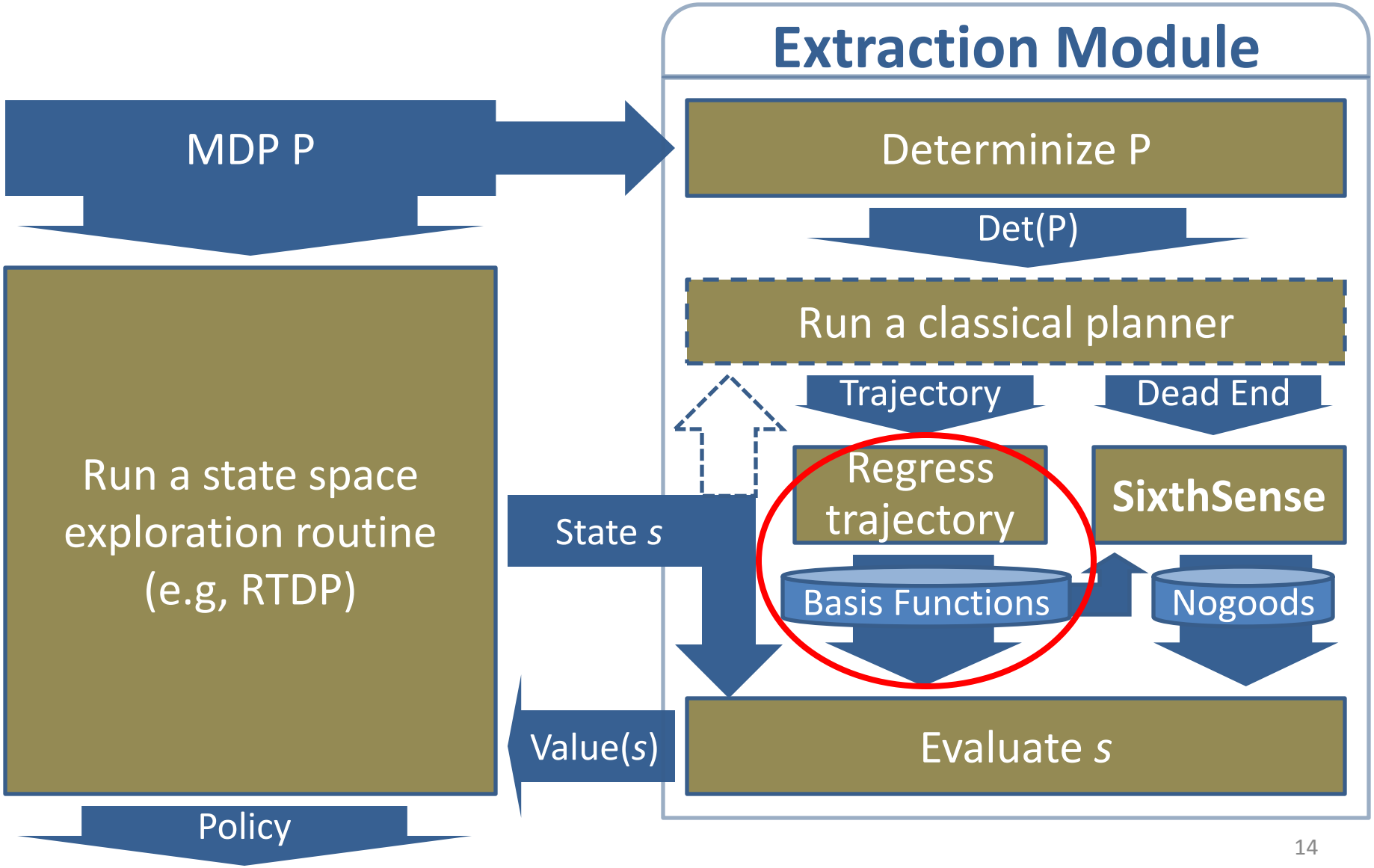
Generating Trajectories



Generating Trajectories



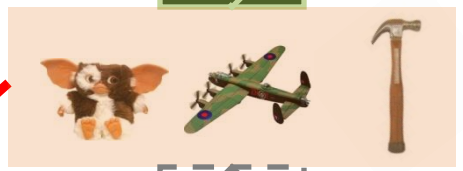
Computing Basis Functions



Regressing Trajectories

basis functions

basis function guarantees goal is reachable from s

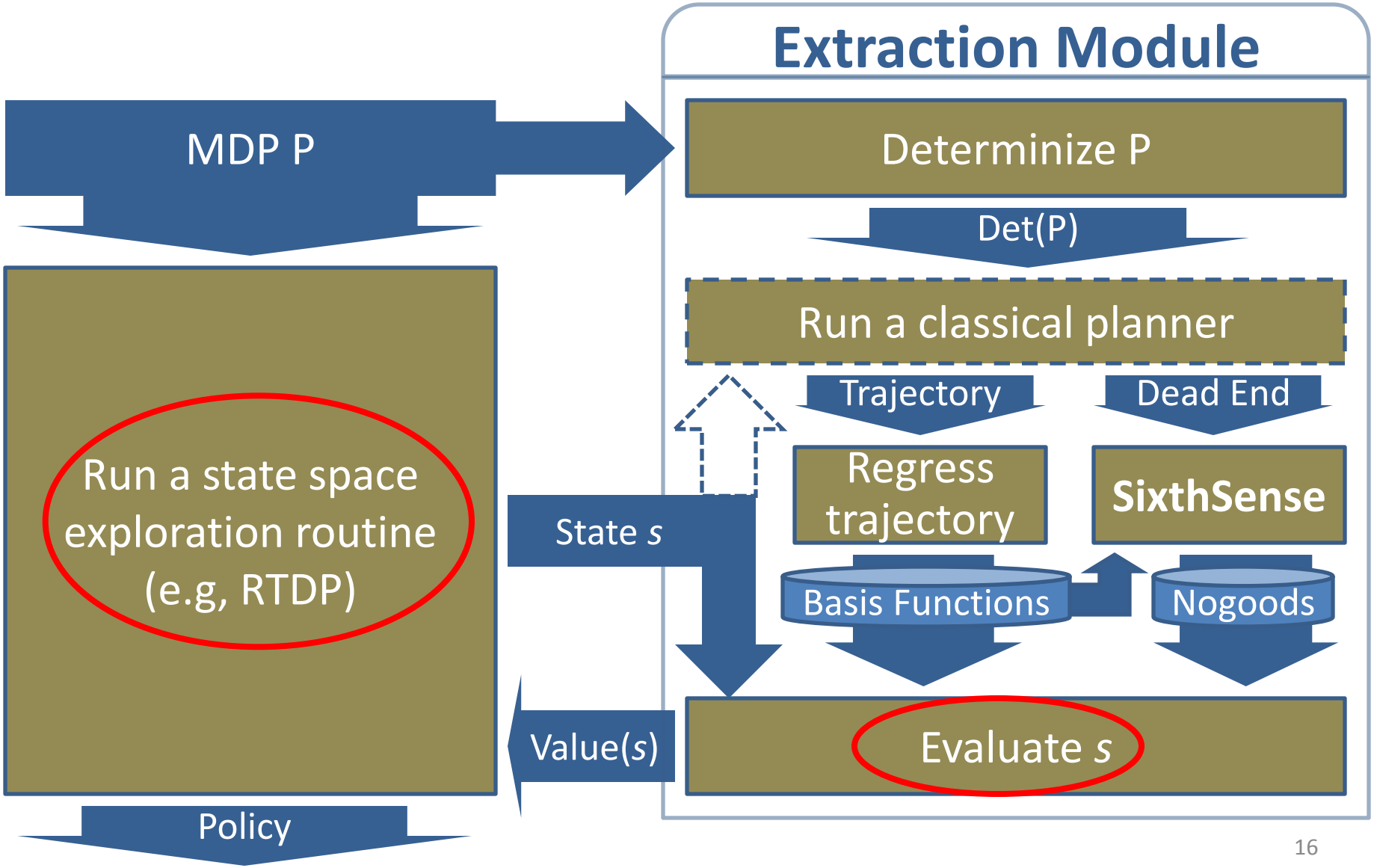


= 1

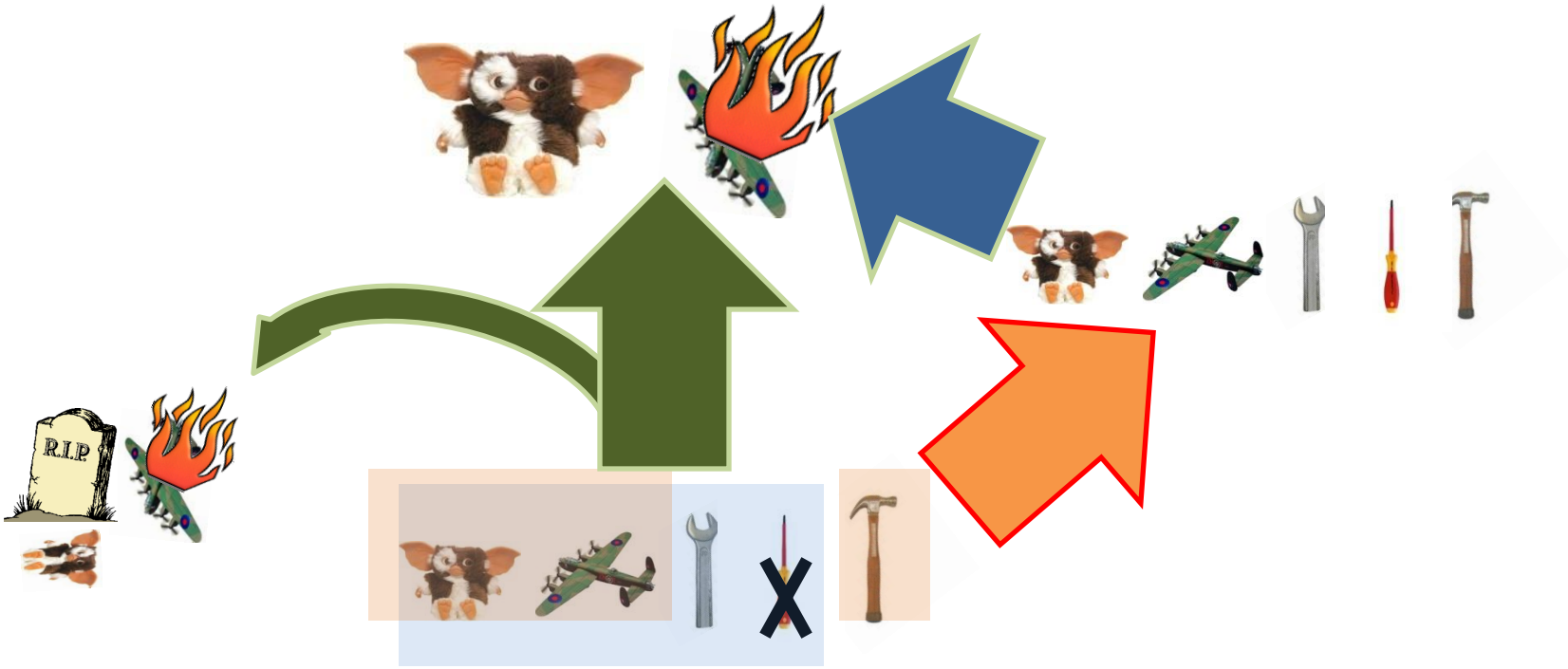
Initial weights

= 2

Computing Values



Meaning of Basis Function Weights



Want to compute basis function weights so that the blue basis function looks "better" than the pink one!

Value of a Basis Function

- Basis function enables at least one trajectory
  - applicable from all relevant states
- Trajectories combine to form policies
- *Value* of a basis function ~ "quality" of its policies

- Algorithm based on RTDP
  - Learn basis function values
  - Use them to compute values of states

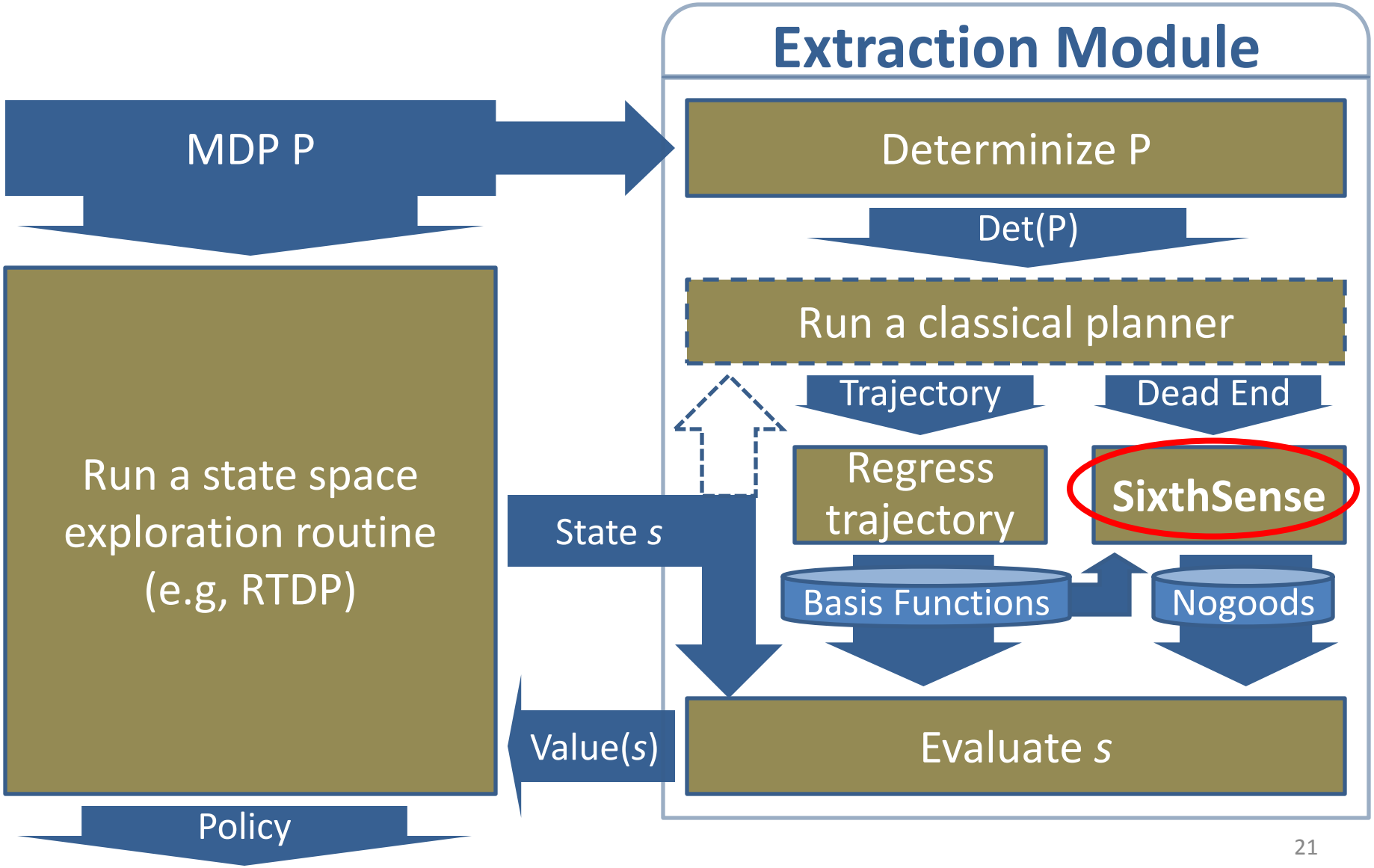
Key Drawback of ReTrASE So Far…

- Dead-end handling expensive
  - expensive to identify: drain on time
  - too many to store: drain on space

Outline

- Motivation
- Running Example
- ReTrASE: Basis Function Generation
- SixthSense: Dead-end Generalization
- Conclusions

Computing Values



Research Question

Can we devis  **dentification procedure fas memoization?**

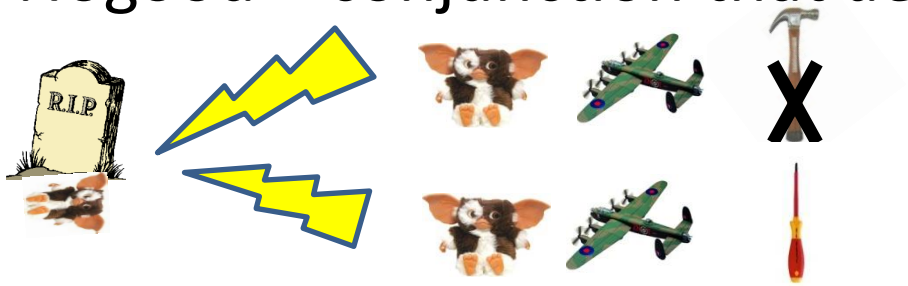
Learns feature combinations whose presence guarantees a state to be a dead end

Nogoods

Nogood

Generate-and-Test Procedure

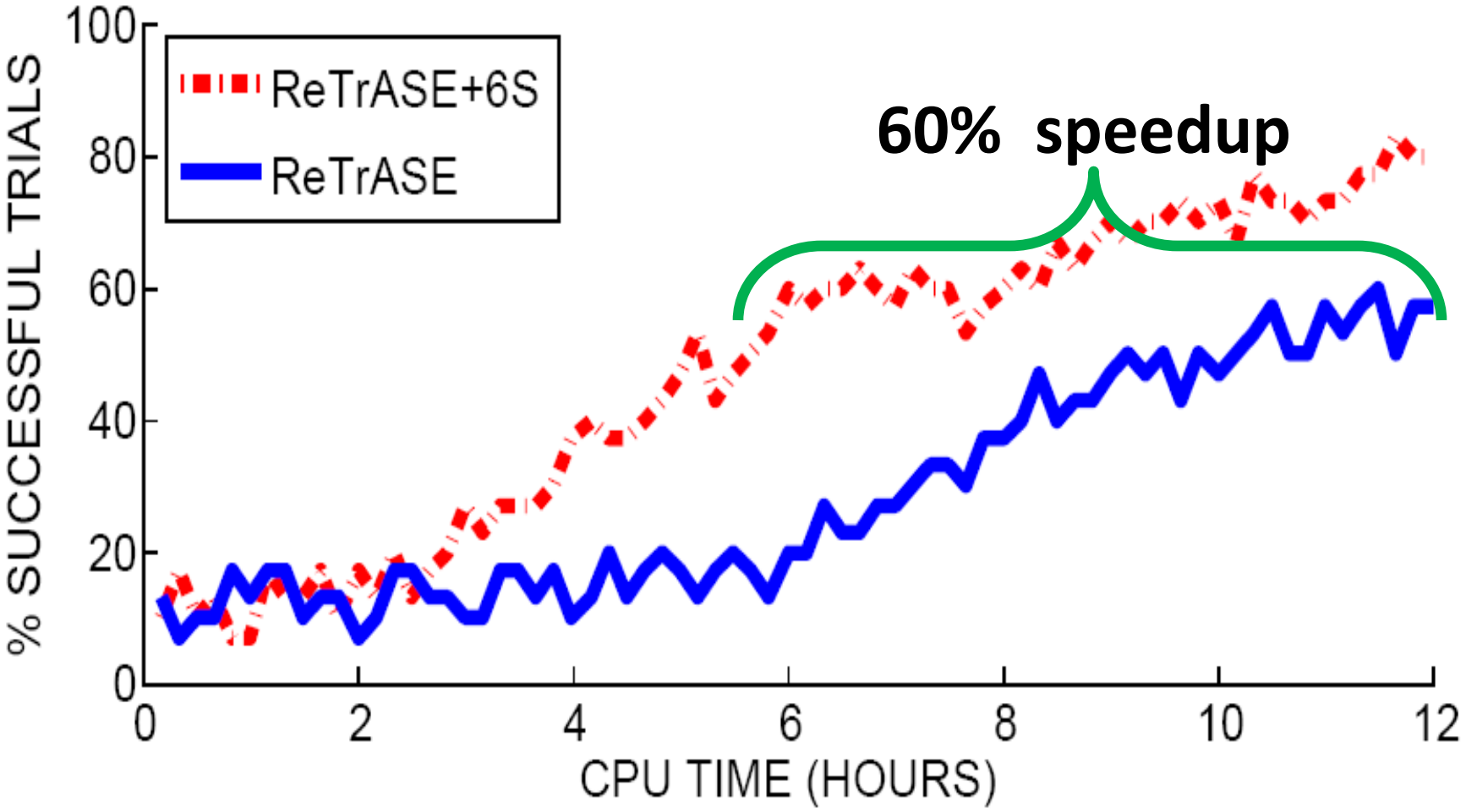
- **Generate a nogood candidate**
  - Key insight: Nogood = conjunction that *defeats* all b.f.s



  - For each b.f., pick a literal that defeats it

- **Test the candidate**
  - Needed for soundness, since we don't know all b.f.s
  - Use the non-relaxed Planning Graph algorithm

Effect of SixthSense on ReTrASE



Experimental Results

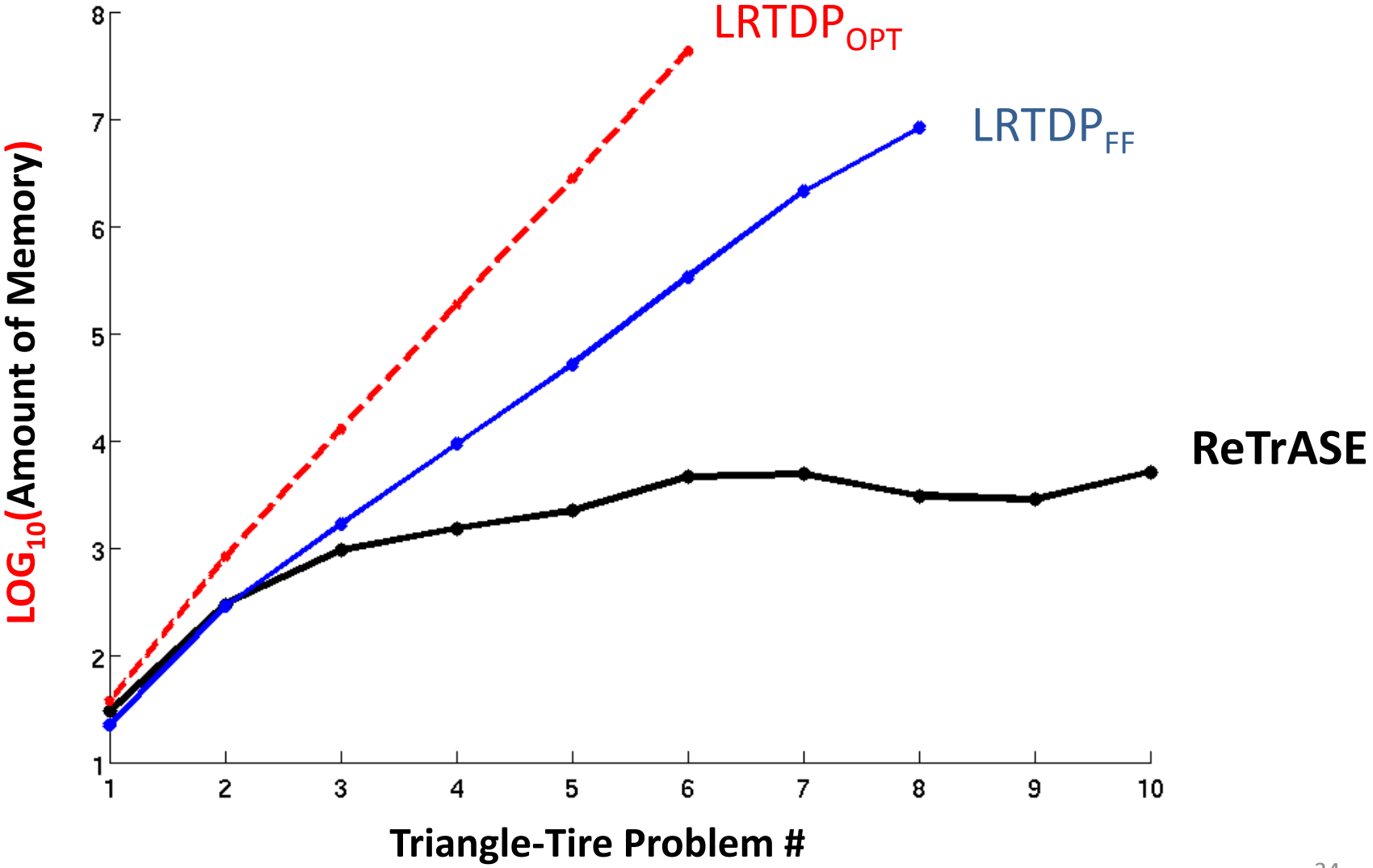
- Criteria:
  - Scalability (vs. VI/RTDP-based planners)
  - Solution quality (vs. IPPC winners)
- Domains: 6 from IPPC-06 and IPPC-08
- Competitors:
  - Best performer on the particular domain
  - Best performer in the particular IPPC
  - LRTDP

The Big Picture

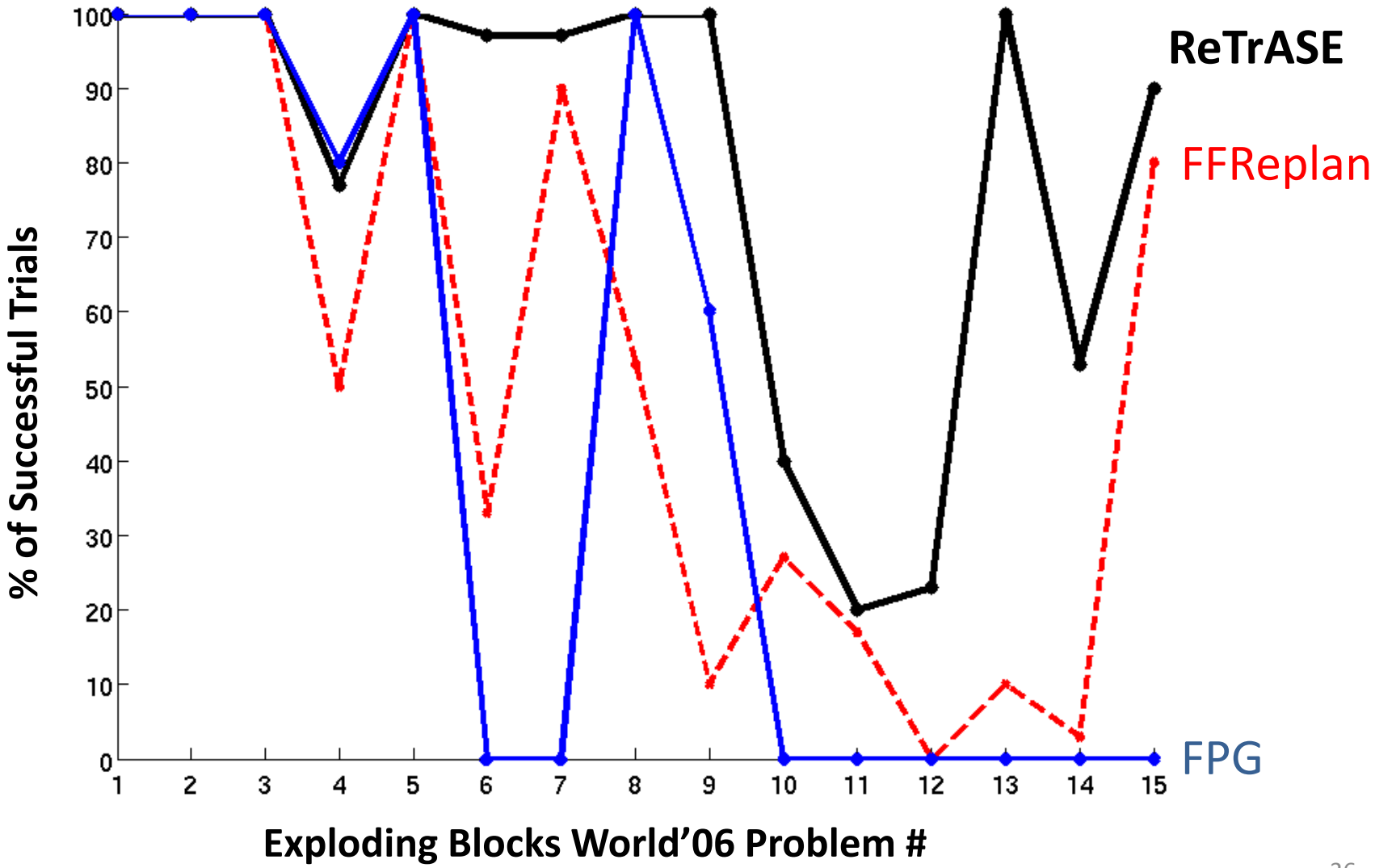
- ReTrASE is vastly more scalable than VI/RTDP-based planners

- ReTrASE typically rivals or outperforms the best-performing planners on IPPC domains

Triangle-Tire: Memory Consumption



Exploding Blocks World: Success Rate



Outline

- Motivation
- Running Example
- ReTrASE: Basis Function Generation
- SixthSense: Dead-end Generalization
- Conclusions

Take Homes

- Novel ideas to learn structure in the domain
- Basis functions
  - Learn by regressing trajectories
  - Represent good structure
  - Generalize across states
- Nogoods
  - Learn inductively; prove using a sound procedure
  - Represent bad structure
  - Generalize across dead-end states

Take Homes

- A novel use of classical planners for MDP algos
  - retains the decision-theoretic nature of MDPs
  - exploits the scalability of classical planners
- Automatic ways to generate basis functions
  - no longer an onus on human designer
  - exploits factored domain model