Conformant Planning:

Symbolic Model Checking + Heuristic Search

Alessandro Cimatti

Istituto per la Ricerca Scientifica e Tecnologica, Trento, Italy

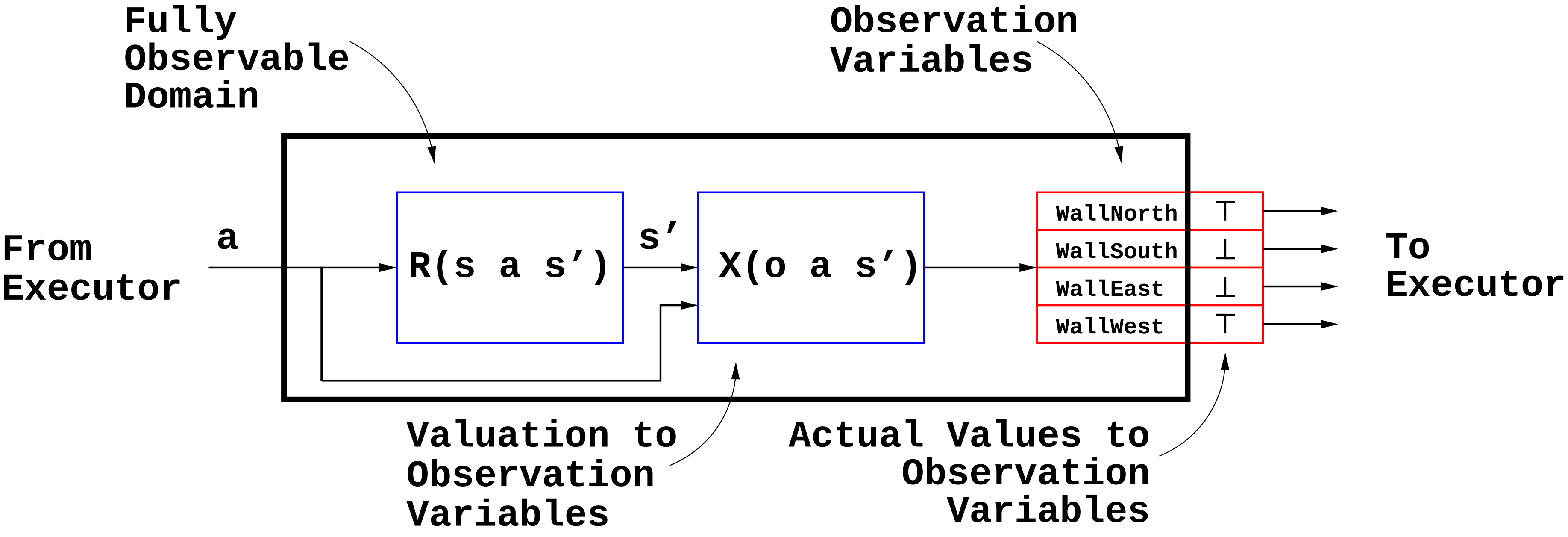
cimatti@irst.itc.it

Joint work with Piergiorgio Bertoli and Marco Roveri.

Introduction

Non-classical planning in real world domains:

- Initial condition is incompletely specified;
- Action effects may be nondeterministic and unpredictable;
- Run-time observability can be limited.



Remark: a plan can be associated with many execution traces.

Planning via Symbolic Model Checking

Basic intuitions:

- planning domain as finite-state automaton;
- planning as state-space exploration;
- symbolic, BDD-based model checking techniques;

Some results:

- BDD-based generation of Universal plans
- Temporally extended goals
- conditional planning under partial observability.

In this talk, we tackle the problem of Conformant Planning:

- find a (sequential) plan, guaranteed to reach the goal
- from an incompletely specified initial condition
- despite nondeterministic action effects
- no information can be acquired at run time (no observability);

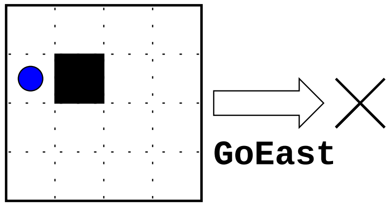
Structure of the talk

- ☞ Conformant Planning as Search in the Space of Beliefs
- ☐ Efficient Conformant Planning
  - = Symbolic BDD-based Model Checking techniques
  - + Heuristic Search in the Space of Beliefs
- ☐ Experimental Evaluation
- ☐ Related Work
- ☐ Conclusions and Future Work

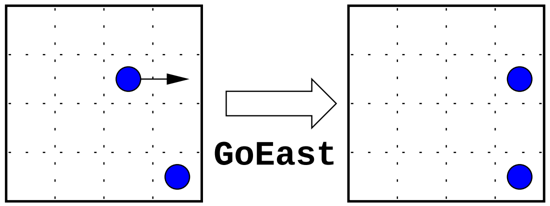
Nondeterministic Domains

We consider a general model of nondeterministic domains, where actions have

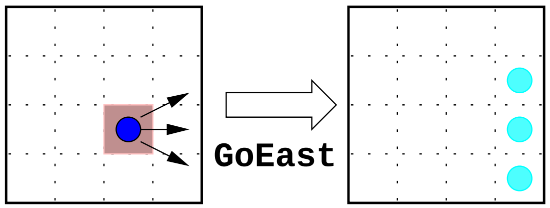
- Preconditions:



- Conditional effects:



- Uncertain effects:

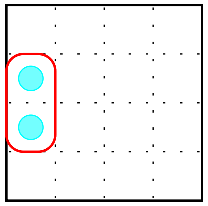


Domains as finite state automaton: $\mathcal{D} = \langle \mathcal{P}, \mathcal{S} \subseteq Pow(\mathcal{P}), \mathcal{A}, \mathcal{R} \subseteq \mathcal{S} \times \mathcal{A} \times \mathcal{S} \rangle$

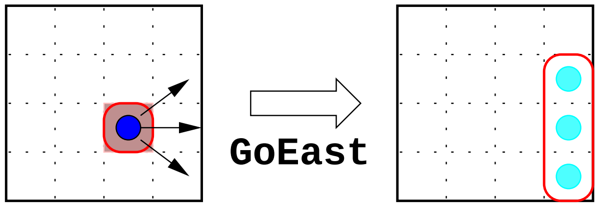
Representing Incomplete Information with Belief States

Incompleteness of information can be due to:

- Incompletely specified initial condition:



- Nondeterministic action effects:

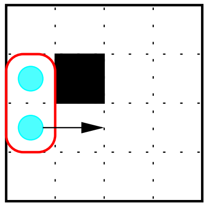


Indistinguishable states are collected into sets called Belief States.

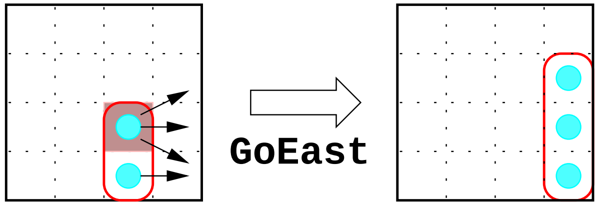
The Belief Space is the set of all belief states, i.e. $Pow(\mathcal{S})$.

Actions in Belief Space

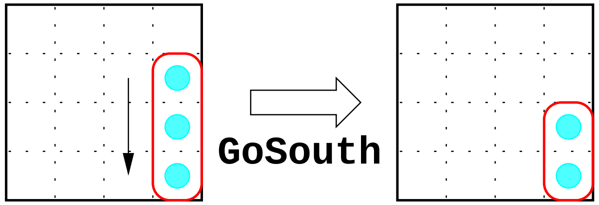
- Action applicability must be guaranteed



- Action execution gathers all effects



- Action execution can reduce uncertainty



Conformant Planning

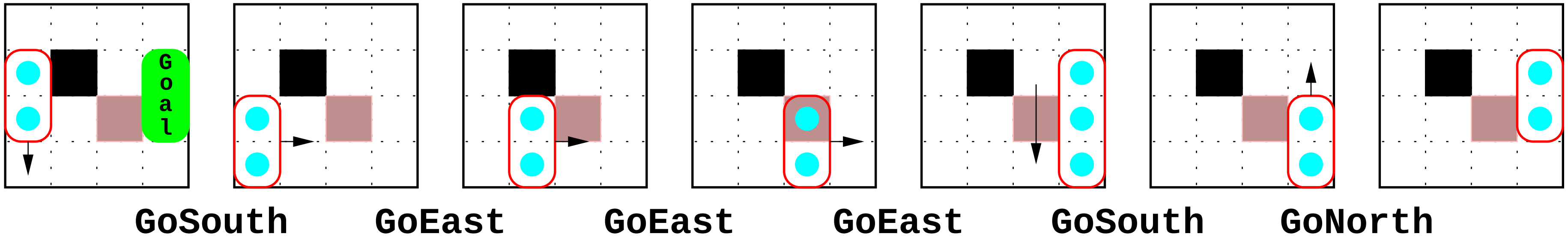
Conformant Planning Problem:

- a set of goal states $\mathcal{G} \subseteq \mathcal{S}$
- a set of initial states $\mathcal{I} \subseteq \mathcal{S}$

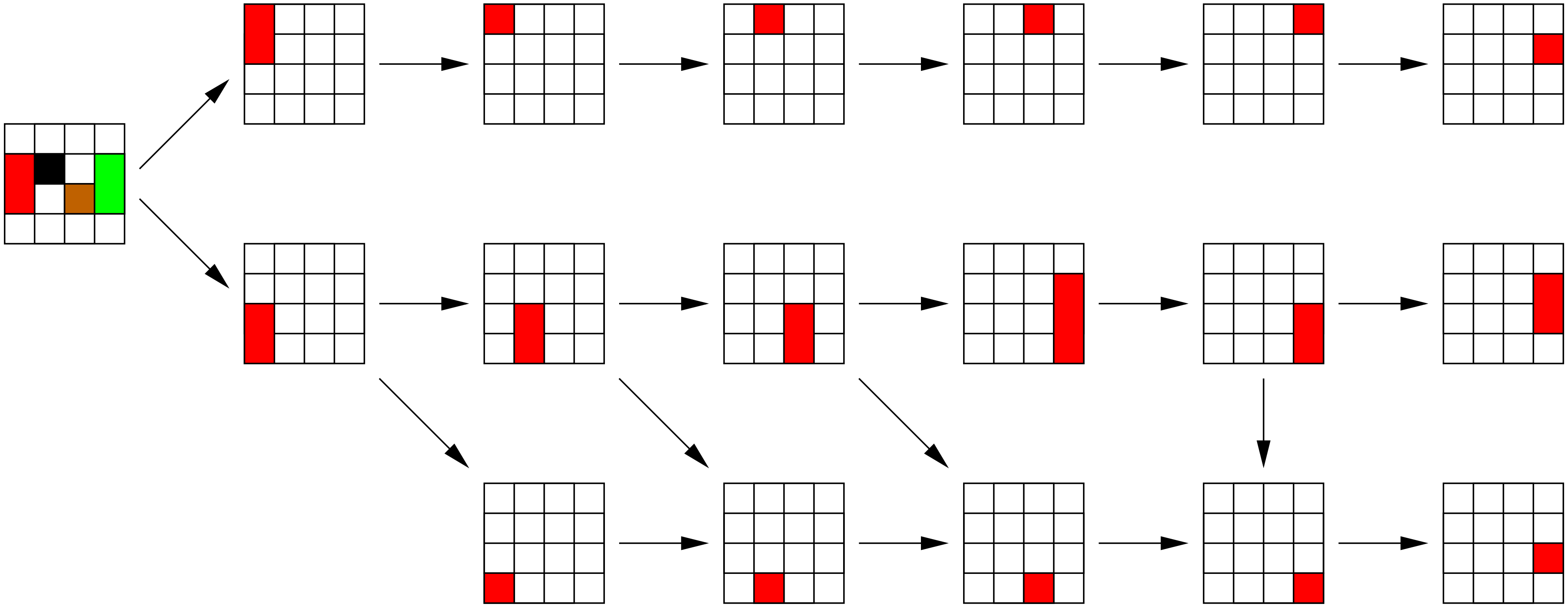
Conformant Plan:

- never violates action applicability;
- all resulting states are goal states.

Example:



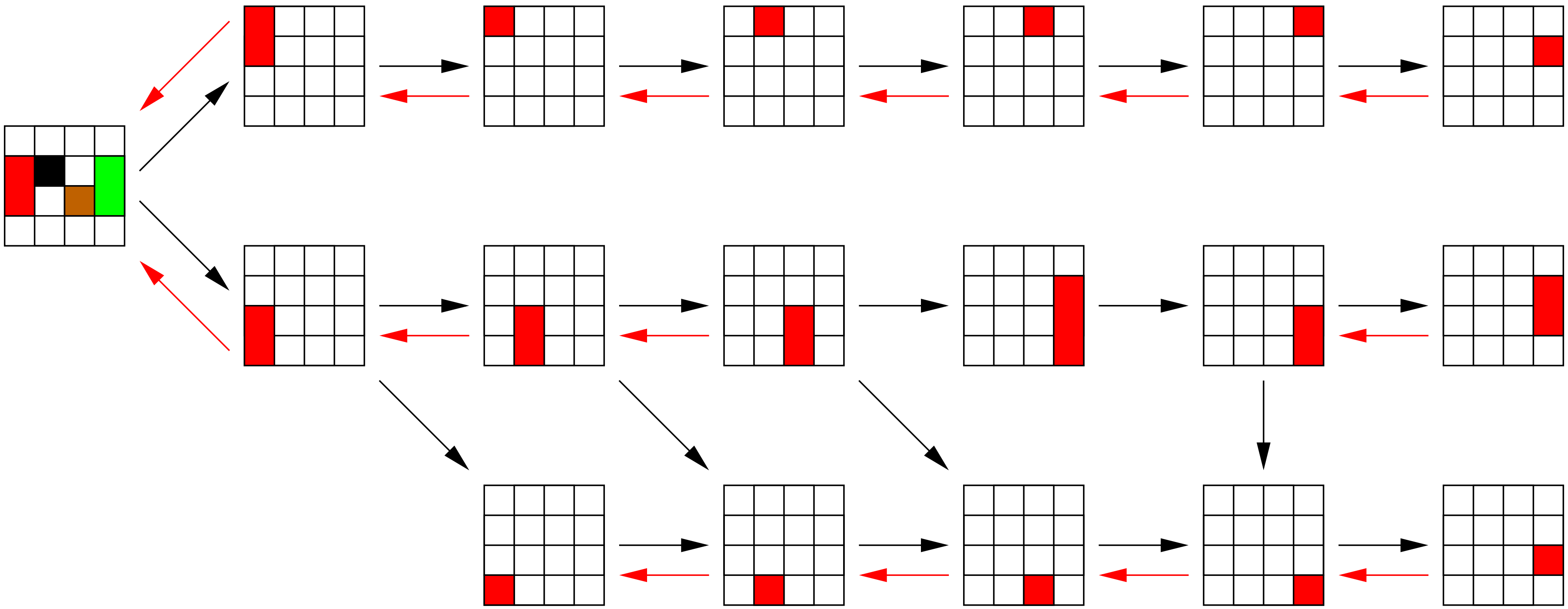
The Search Space



Remark:

- the search space is $Pow(\mathcal{S})$
- $\mathcal{S}$ contains 15 states,
- $Pow(\mathcal{S})$ contains up to 32767 belief states!

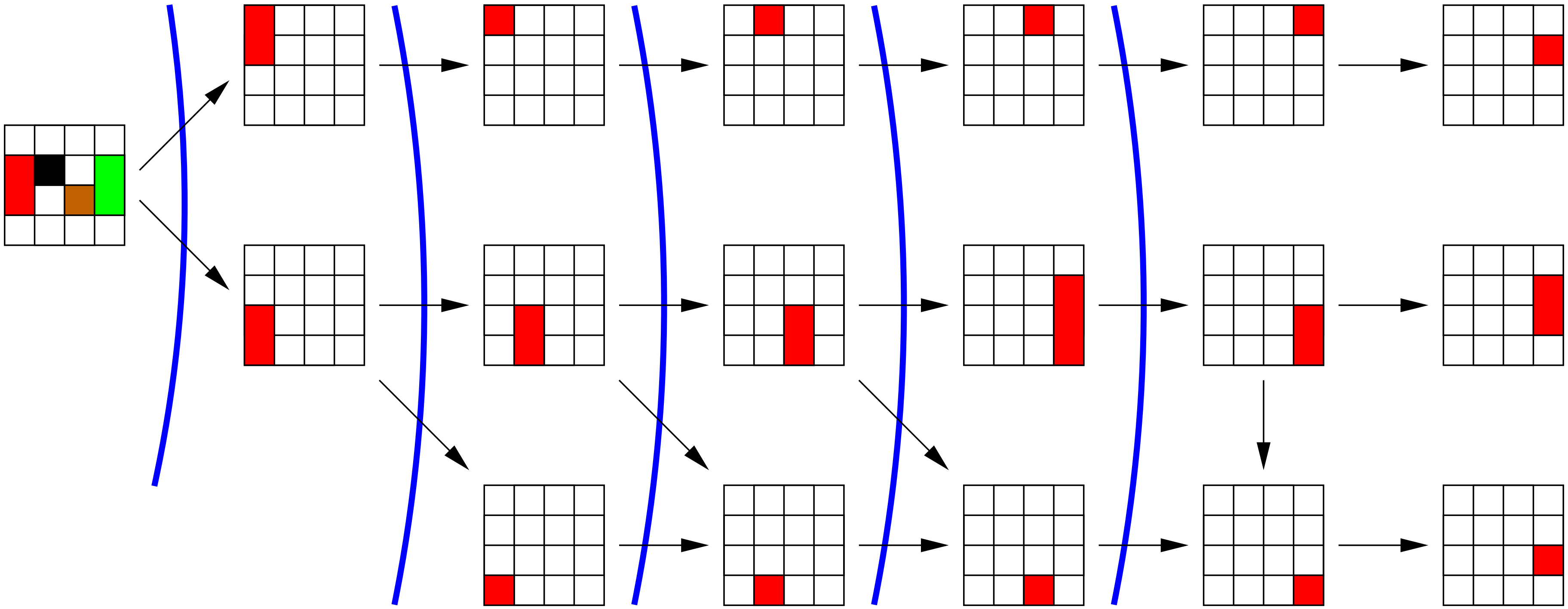
The Heuristic Search Approach



Bonet and Geffner [AIPS2000]: Heuristic Search in Belief Space

+ Goal-directed search
– Explicit Representation of Belief States: $\{s_1, \ldots, s_n\}$
– Expansion of Belief States via Enumeration of Transitions

The Symbolic BFS Approach



Cimatti and Roveri [JAIR2000]: Breadth-first search with BDD-based techniques

– Blind search

+ BDD-based representation of Belief States

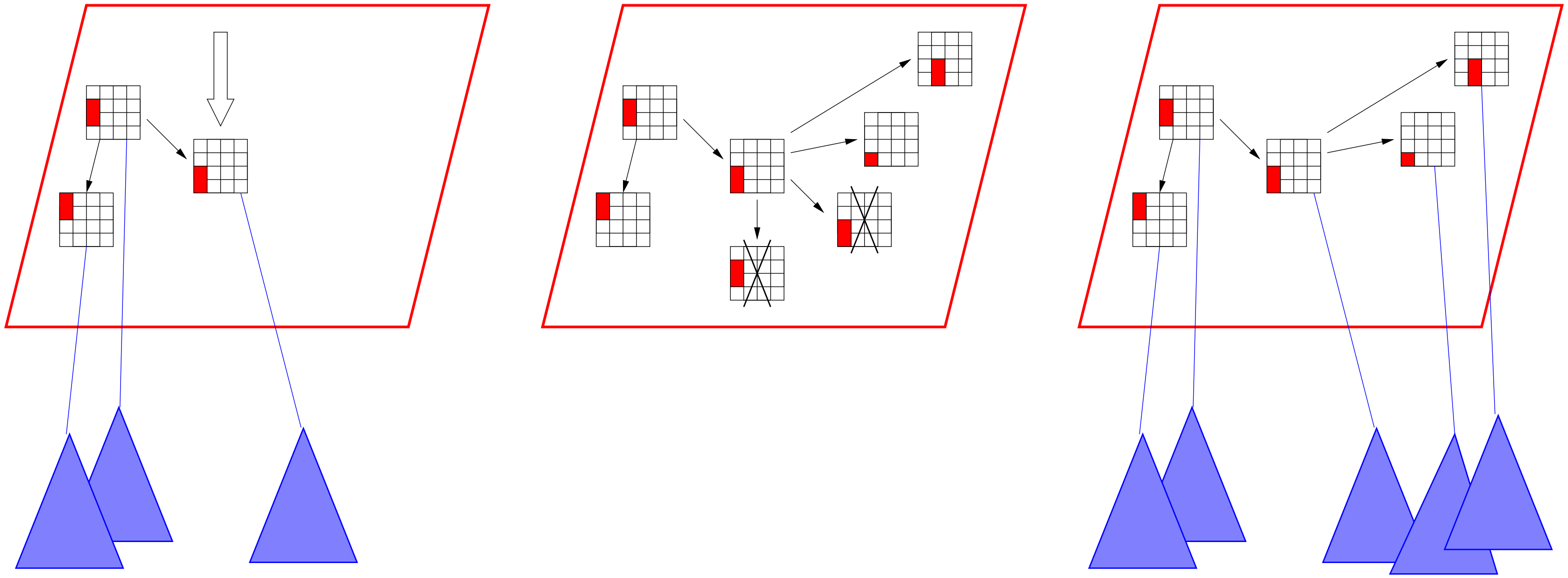
+ Symbolic, non-enumerative expansion of transitions

Structure of the talk

- ✓ Conformant Planning as Search in the Belief Space
- ☞ Efficient Conformant Planning
  - = Symbolic BDD-based Model Checking techniques
  - \+ Heuristic Search in the Space of Beliefs
- ☐ Experimental Evaluation
- ☐ Related Work
- ☐ Conclusions and Future Work

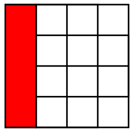
BDD-based Search in Belief Space: Intuitions

- Represent each belief state as a BDD;
  - + Efficient storage of large state spaces
  - + Symbolic expansion avoids transition enumeration
- Search as iterated expansion of “best” belief states.
  - + Goal directed search

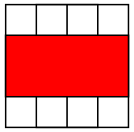


Belief States

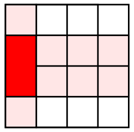
X = 0

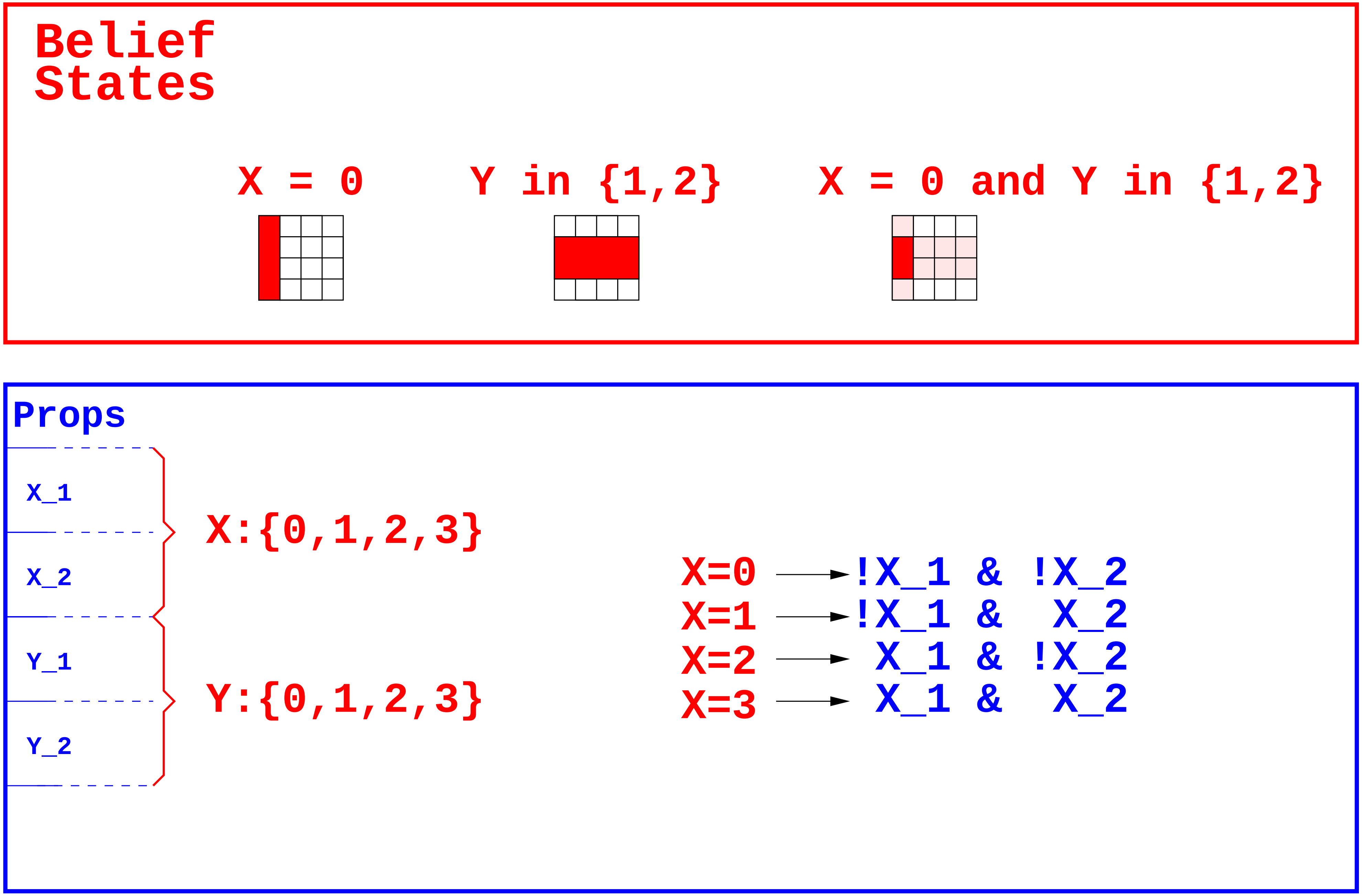


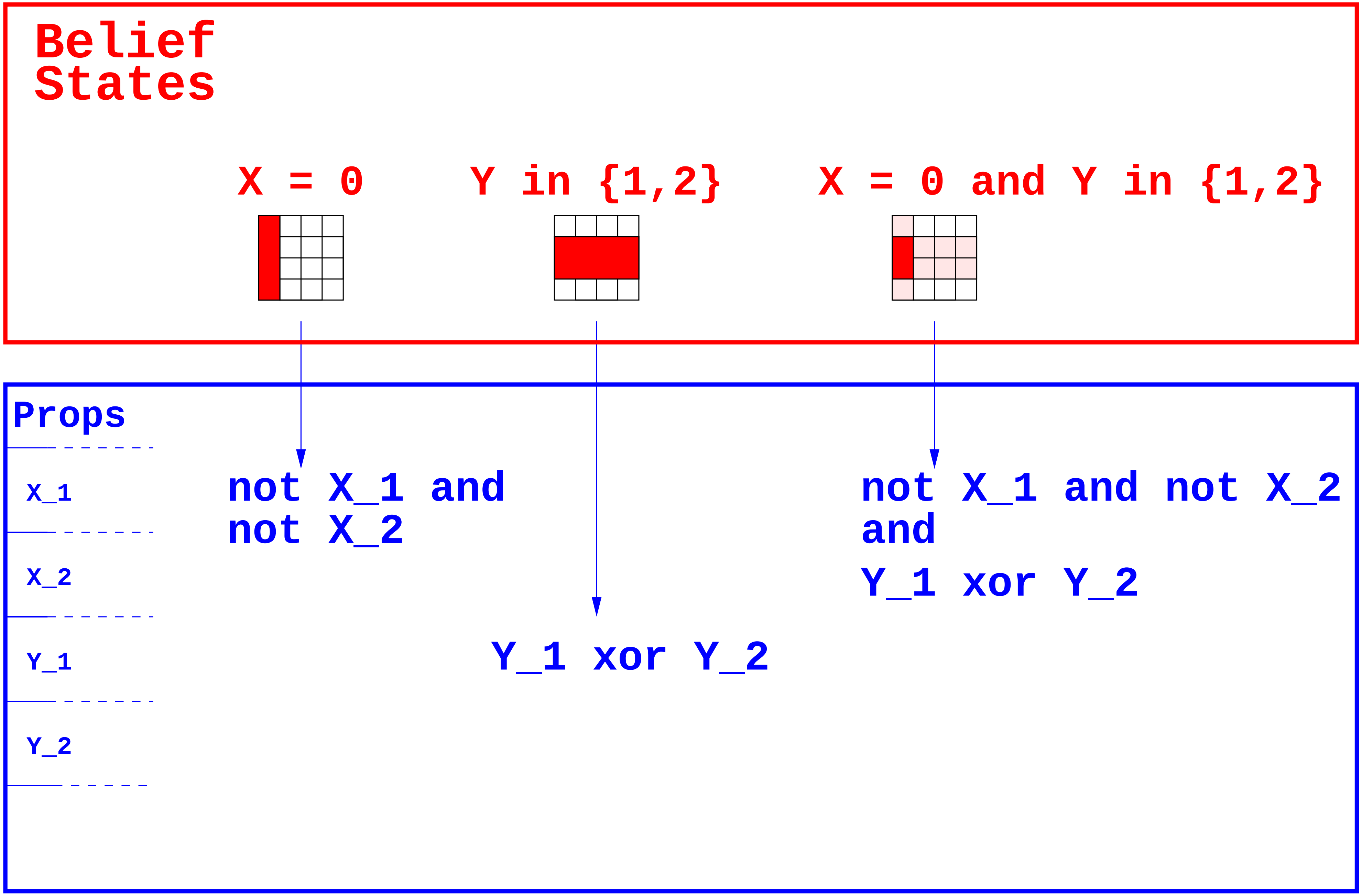
Y in {1,2}

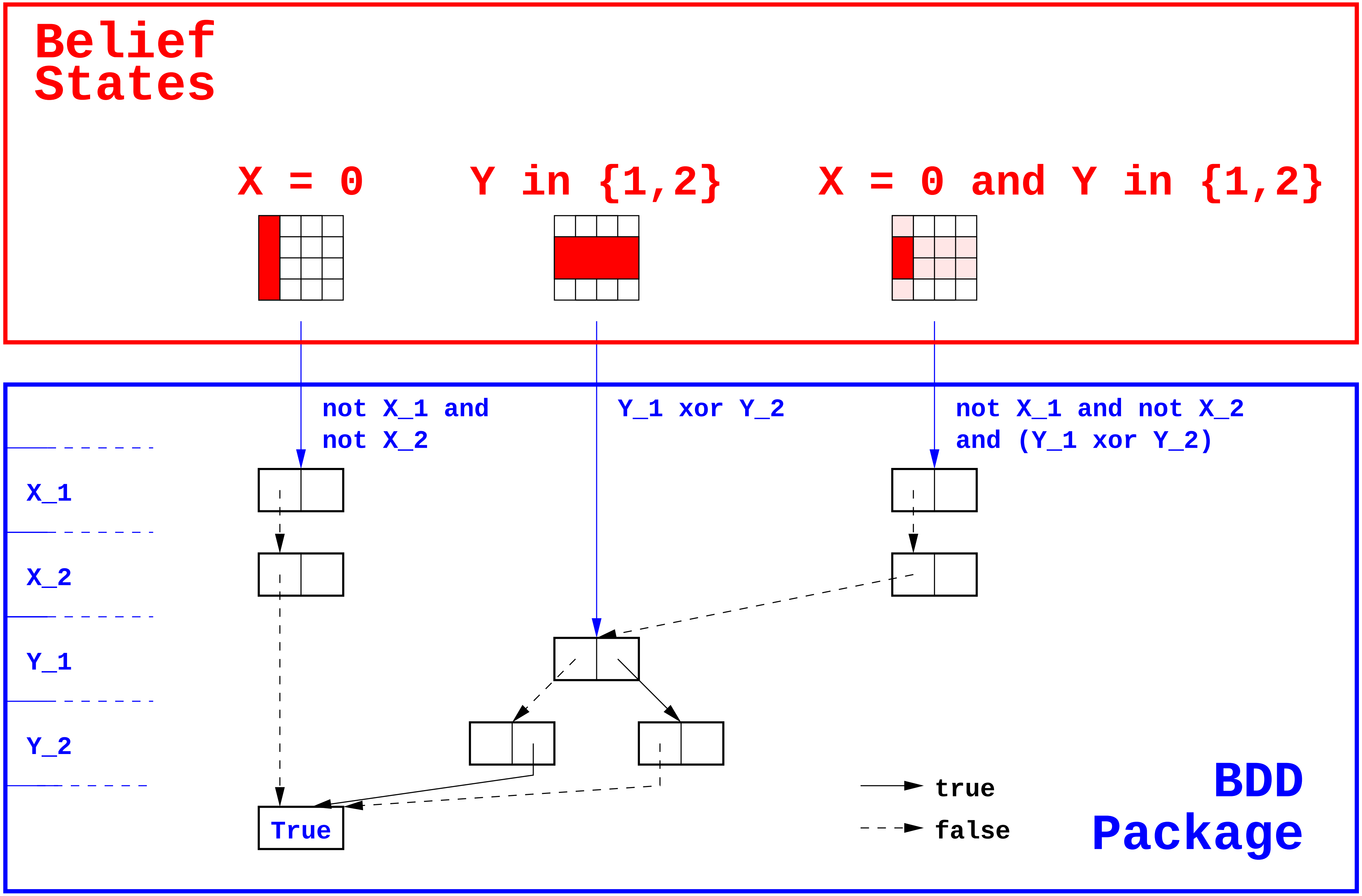


X = 0 and Y in {1,2}




Belief States
X = 0
Y in {1,2}
X = 0 and Y in {1,2}
Props
X_1
X_2
Y_1
Y_2
X:{0,1,2,3}
Y:{0,1,2,3}
X=0 → !X_1 & !X_2
X=1 → !X_1 & X_2
X=2 → X_1 & !X_2
X=3 → X_1 & X_2


Belief States
X = 0
Y in {1,2}
X = 0 and Y in {1,2}
Props
X_1
X_2
Y_1
Y_2
not X_1 and not X_2
Y_1 xor Y_2
not X_1 and not X_2 and Y_1 xor Y_2

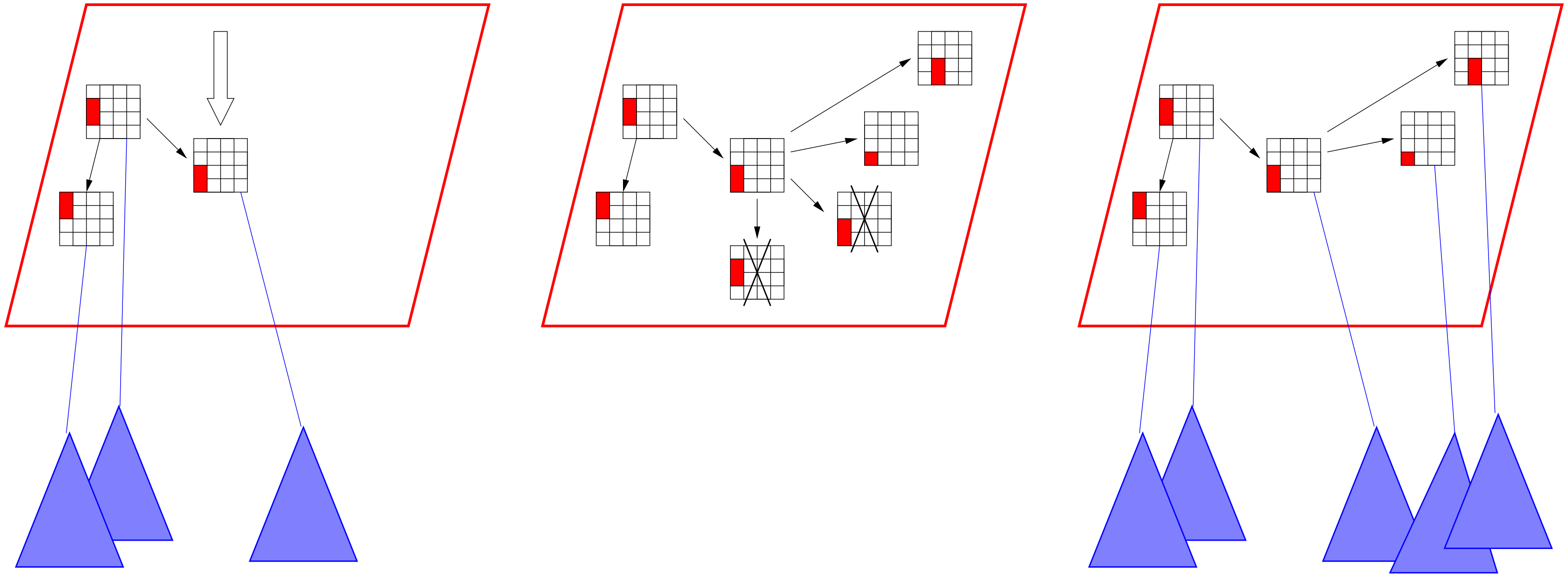

Belief States
X = 0
Y in {1,2}
X = 0 and Y in {1,2}
not X_1 and
not X_2
Y_1 xor Y_2
not X_1 and not X_2
and (Y_1 xor Y_2)
X_1
X_2
Y_1
Y_2
True
true
false
BDD Package

BDD-based Search in Belief Space

Each belief state is associated to a BDD (unique thanks to canonical form).

Visited belief states hashed with BDD pointer used as key.

Search as iterated expansion of “best” belief states.



Key problem: How can we perform the expansion efficiently?

BDD-based Expansion of a Belief State

Key step is the efficient expansion of Bs, i.e. the computation of the set

$$\{\langle \alpha_1 \,.\, \mathit{Exec}[\alpha_1](Bs)\rangle, \ldots, \langle \alpha_k \,.\, \mathit{Exec}[\alpha_k](Bs)\rangle\}$$

where α_i is applicable in Bs, and $\mathit{Exec}[\alpha_i](Bs)$ is new.

The effect of actions is represented by the BDD $\mathcal{R}(\boldsymbol{x}, \boldsymbol{\alpha}, \boldsymbol{x}')$, where $\boldsymbol{x}$, $\boldsymbol{\alpha}$ and $\boldsymbol{x}'$ are the current state, action and next state variables.

In terms of BDDs, the steps to compute the Expansion of Bs are the following:

- compute the BDD for all the actions applicable in Bs, i.e.

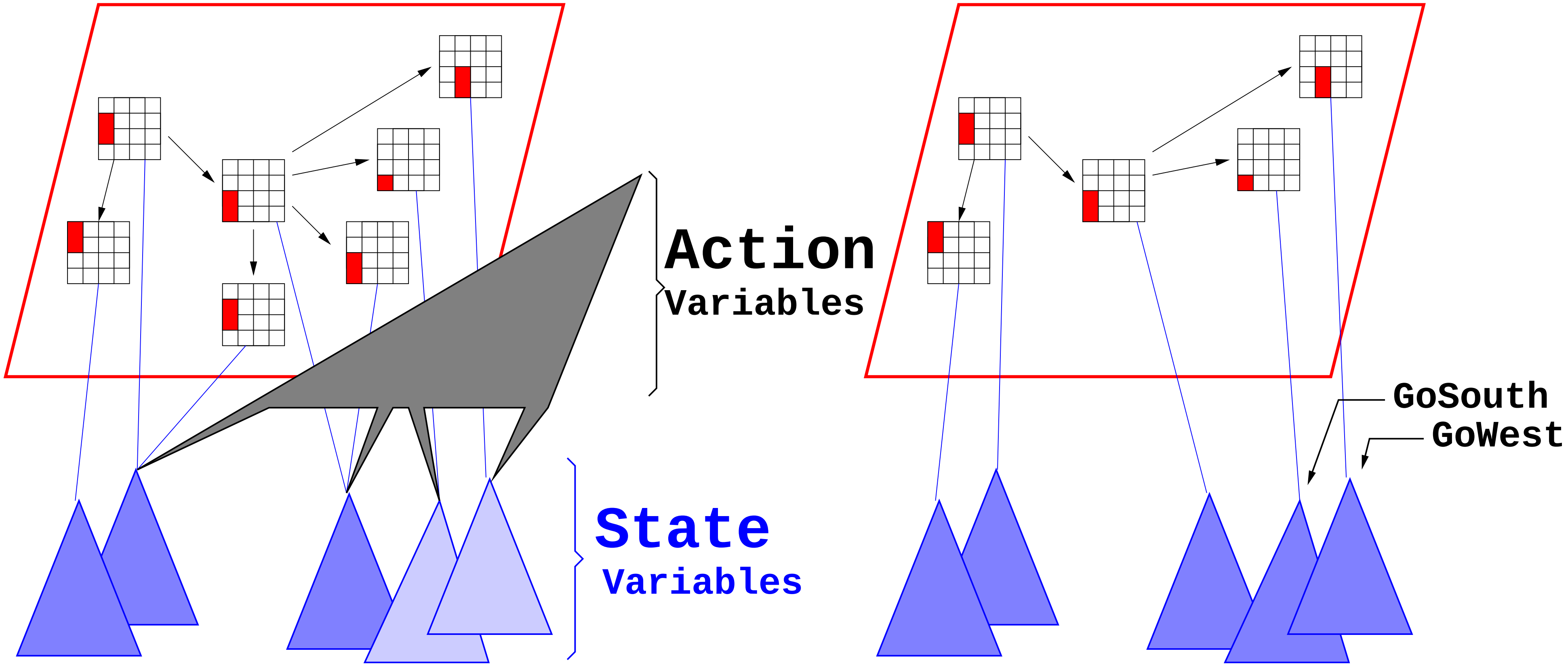
$$\textsc{ApplActs}_{Bs}(\boldsymbol{\alpha}) \doteq \forall \boldsymbol{x}.(Bs(\boldsymbol{x}) \rightarrow \exists \boldsymbol{x}'.\mathcal{R}(\boldsymbol{x}, \boldsymbol{\alpha}, \boldsymbol{x}'))$$

- compute the BDD

$$(\exists \boldsymbol{x}.(Bs(\boldsymbol{x}) \wedge \textsc{ApplActs}_{Bs}(\boldsymbol{\alpha}) \wedge \mathcal{R}(\boldsymbol{x}, \boldsymbol{\alpha}, \boldsymbol{x}')))[\boldsymbol{x}'/\boldsymbol{x}]$$

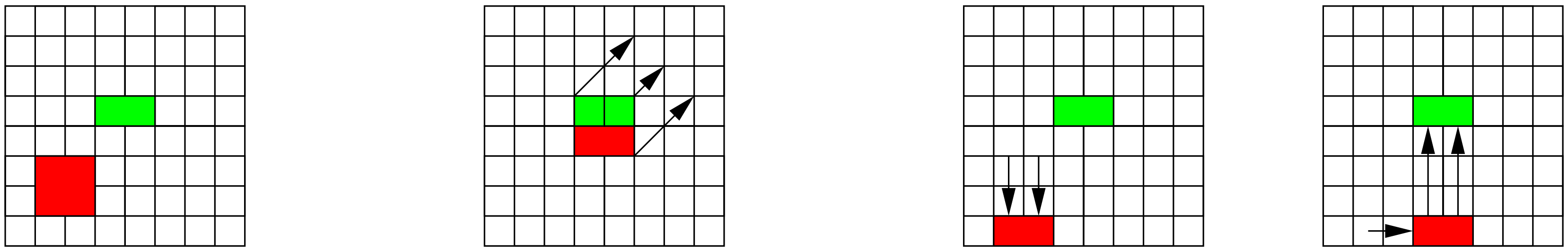
  that depends on $\boldsymbol{x}$ and $\boldsymbol{\alpha}$, and represents all the “applied action-possible result” pairs

Pruning the Expansion of a Belief State



Properties of Heursitic-Symbolic Conformant Planning

- Search can be performed either forward or backward.
- Termination: follows from finiteness of search space.
- If the problem is solvable, a conformant plan is returned.
- If the problem is not solvable, Failure is returned
  - cfr. bounded-depth conformant planning
- Fully amenable to the use of heuristic functions.
  - Ongoing work: smarter heuristics based on planning knowledge acquisition.



- Experimental evaluation with pure heuristic on "degree of uncertainty" (cardinality of belief state);

Structure of the talk

- ✓ Conformant Planning as Search in the Belief Space
- ✓ Efficient Conformant Planning
  - = Symbolic BDD-based Model Checking techniques
  - + Heuristic Search in the Space of Beliefs
- ☞ Experimental Evaluation
- ☐ Related Work
- ☐ Conclusions and Future Work

Experimental Analysis

Heuristic-Symbolic Conformant Planning (HSCP) implemented in MBP:

- general planner for nondeterministic domains
- built on top of the NuSMV BDD-based symbolic model checker
- download at `http://sra.itc.it/tools/mbp/`

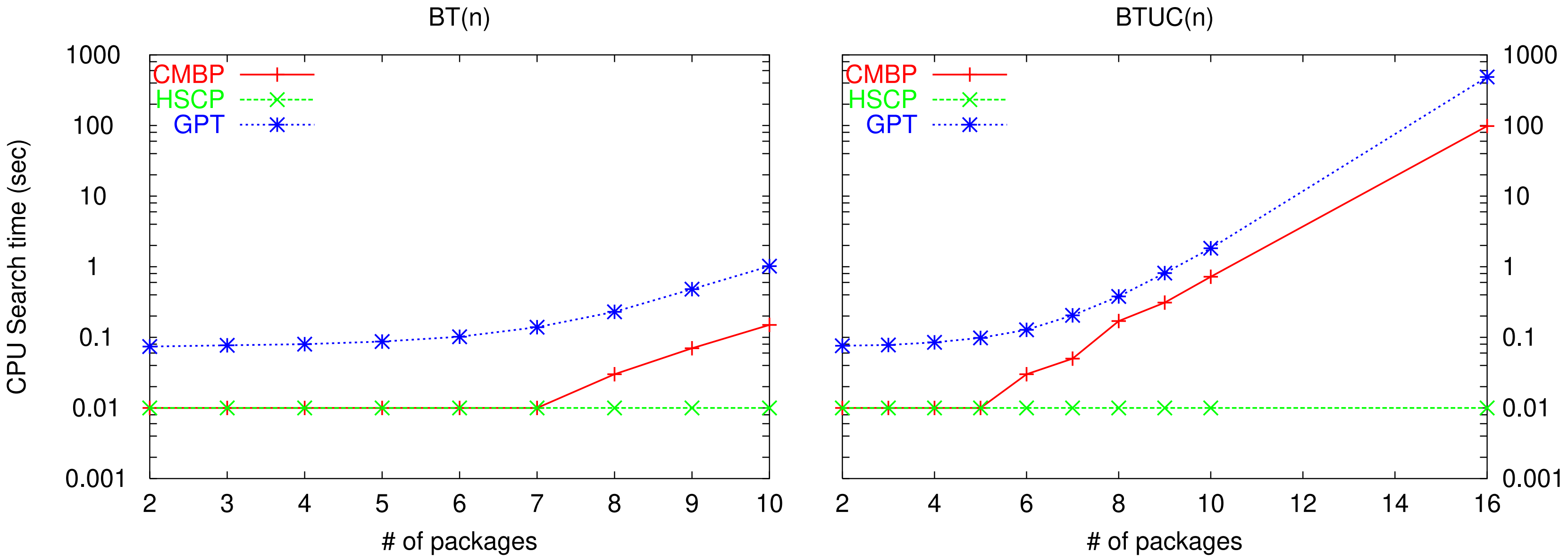
Compared with:

- GPT (Bonet and Geffner): heuristic search (non symbolic)
- CMBP (Cimatti and Roveri): breadth-first BDD-based search

Experiments:

- cover all examples from [JAIR2000];
- examples parameterized on size, and degree of uncertainty (e.g. # of initial states);
- cut-off on CPU time fixed to 2hrs, memory limit fixed to 128Mb;
- Only *search times* are reported (but automaton construction < 0.5 sec)

Basic BT Problems

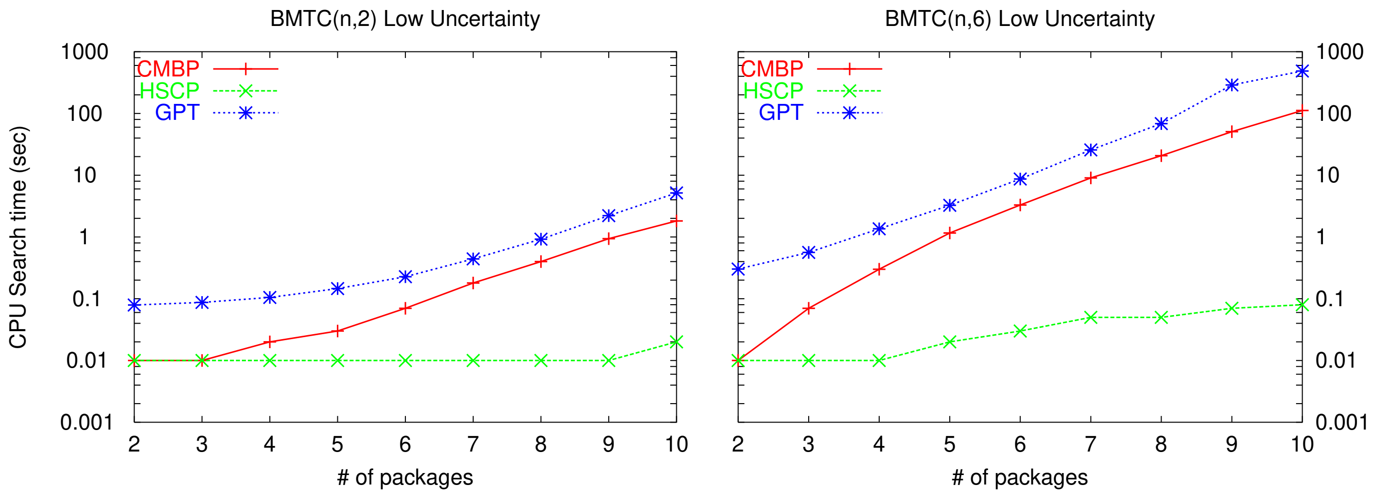


Features:

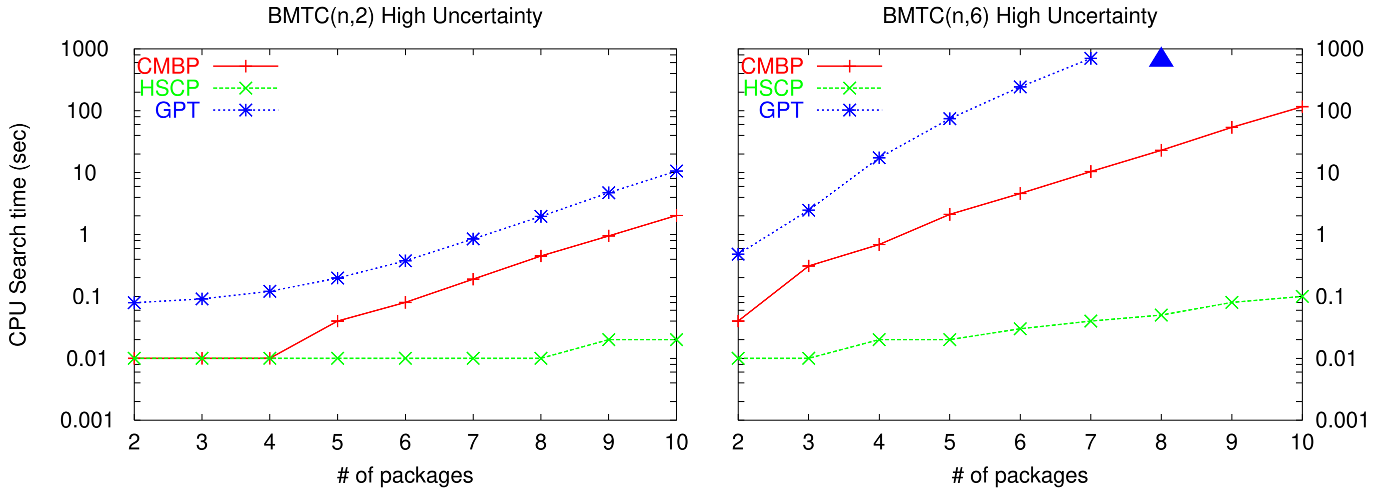
- First problem: highly parallelizable, rather easy
- Second problem: serial, significant branching rate

Multiple BT problems

Low uncertainty:

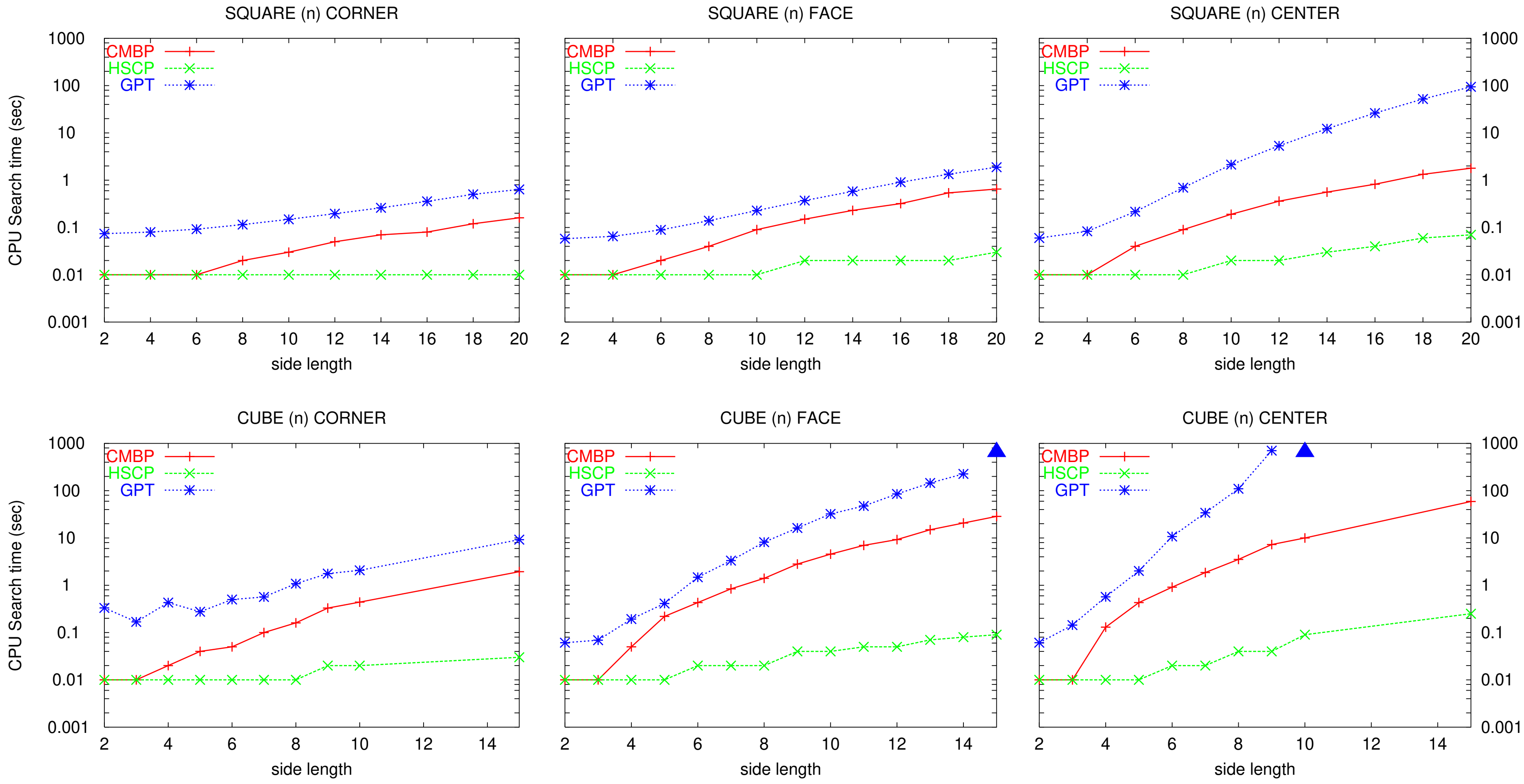


High uncertainty:

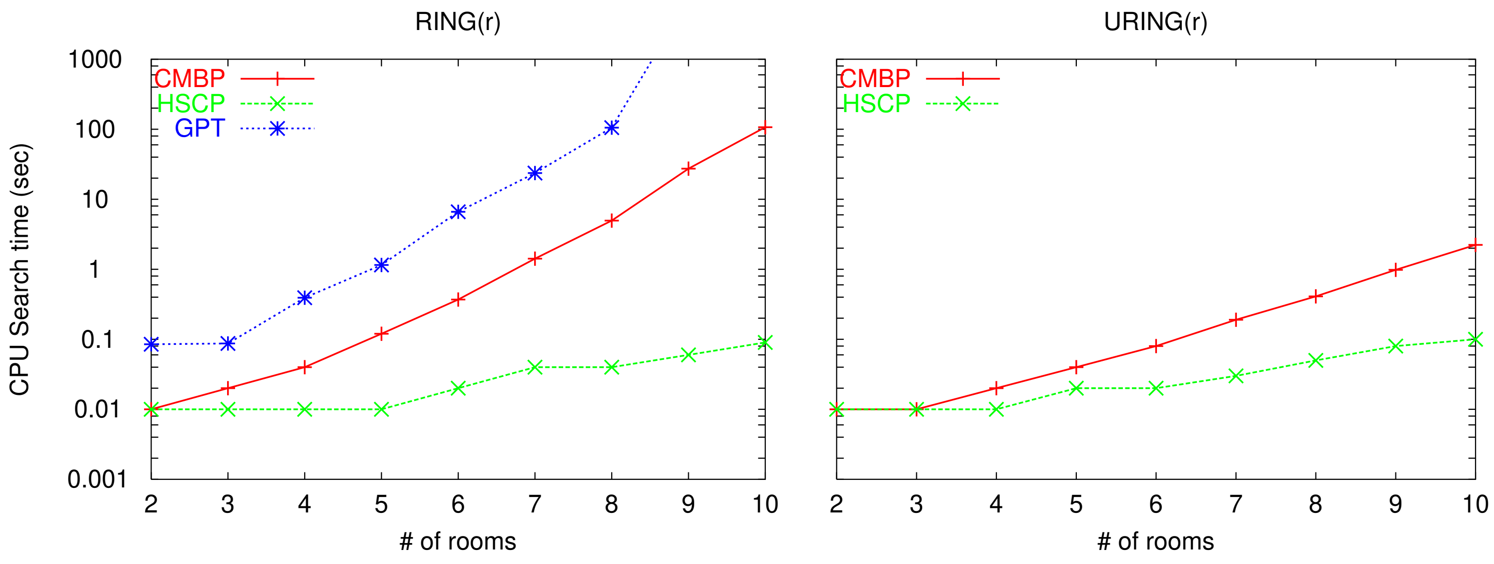


Square/Cube Navigation

Effectiveness of Heuristic Function:



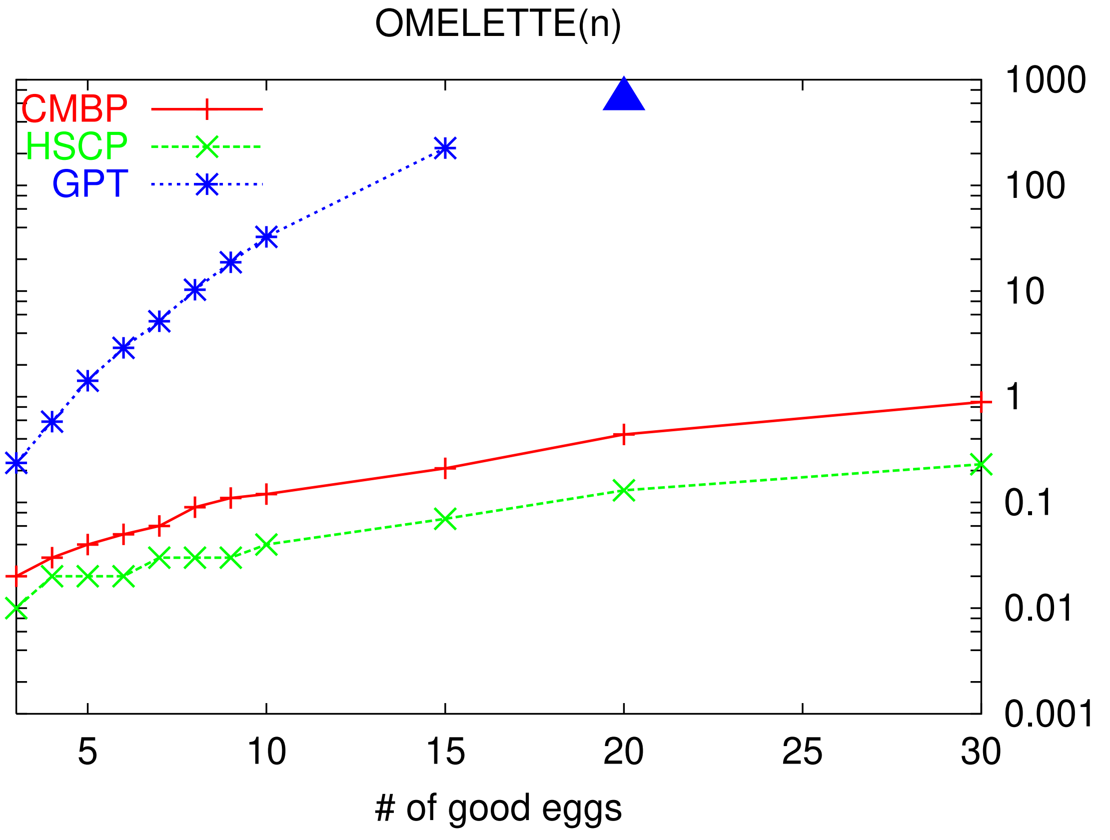
High uncertainty problems



- First problem: high number of intial states
- Second problem: high branching factor

Convergence for unsolvable problems

Surprisingly, HSCP also faster than CMBP in detecting usolvability.



Structure of the talk

✓ Conformant Planning as Search in the Belief Space

✓ Efficient Conformant Planning

= Symbolic BDD-based Model Checking techniques

+ Heuristic Search in the Space of Beliefs

✓ Experimental Evaluation

☞ Related Work and Conclusions

Related Work

Other approaches to Conformant Planning:

- CGP (Smith and Weld): graph planning;
- QbfPlan (Rintanen): reduction to QBF satisfiability;
- $\mathcal{C}$-Plan (Castellini, Ferraris, Giunchiglia, Tacchella): SAT based;
- Open world planning (Finzi, Pirri, Reiter): prime implicates computation;
- DLV$^{\mathcal{K}}$ (Eiter, Faber, Leone, Pfeiffer, Polleres): disjunctive datalog;
- Planning as abduction (Herzig, Lang and Maurquis)

Related Problem in Hardware Analysis: Synchronization Sequences

Synchronization Sequence Problem:

- take an unobservable circuit to a known state;
- find a sequence of inputs that, when applied to the circuit, guarantee that a unique state is reached;
- Other BDD-based approaches:
  - Pixley, Jeong and Hachtel, 92; Rho, Somenzi and Pixley, 93;
  - specialized algorithms for synchronization of determinsitic machines;
  - our approach able to tackle test set (but more experiments needed).

Conclusions and Future Work

We have presented a new approach to conformant planning.

- Generality of the approach:
  - incompletely specified initial condition and nondeterministic actions
  - not bounded, discovers if problem is unsolvable
- Heuristic search + Symbolic Model Checking.
  - Allows for goal directed search
  - BDD-based techniques help in storage and expansion of search space
- Experimental evaluation shows significant improvements

Future work:

- Optimization of automaton construction (e.g. partitioning techniques)
- Definition of standard Nondeterministic PDDL for systematic experimental evaluation;
- Investigation of Heuristics for Search in Belief Space.