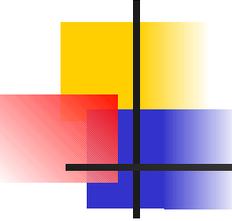


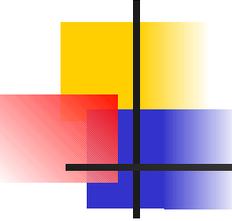
Planning and Verification for Stochastic Processes with Asynchronous Events

Håkan L. S. Younes
Carnegie Mellon University



Introduction

- **Asynchronous** processes are abundant in the real world
 - Telephone system, computer network, etc.
- Discrete-time models are inappropriate for systems with asynchronous events
- We need to go beyond semi-Markov models!

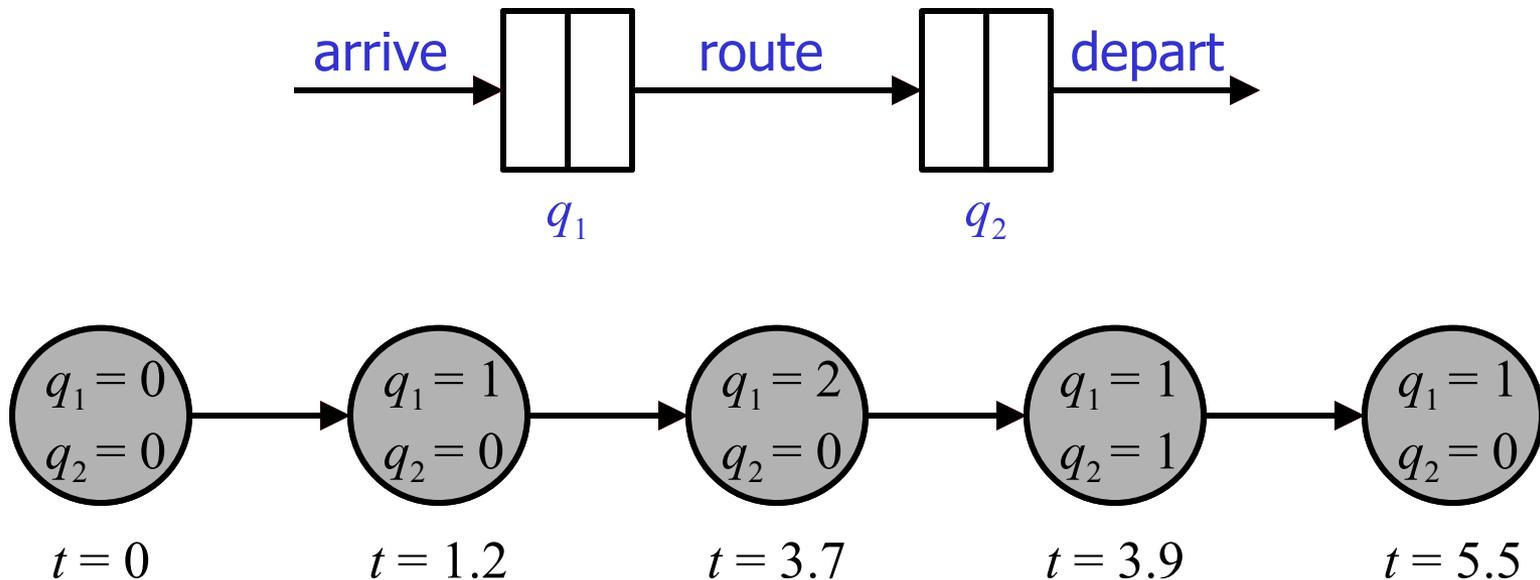


Two Problems

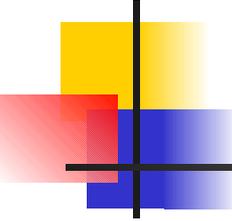
- Model Checking
 - Given a model of an asynchronous system, check if some property holds
- Planning
 - In the presence of asynchronous events, find a control policy that satisfies specified objectives

My thesis research: Provide solution to both problems

Illustrative Example: Tandem Queuing Network



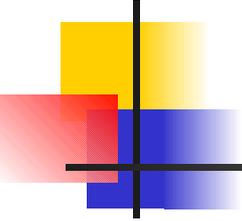
With both queues empty, is the probability less than 0.5 that both queues become full within 5 seconds?



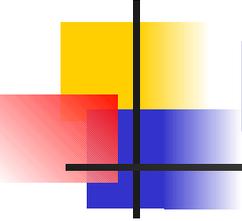
Probabilistic Model Checking

- Given a model M , a state s , and a property φ , does φ hold in s for M ?
 - Model: stochastic discrete event system
 - Property: probabilistic temporal logic formula

Probabilistic Model Checking: Example



- With both queues empty, is the probability less than 0.5 that both queues become full within 5 seconds?
 - State: $q_1 = 0 \wedge q_2 = 0$
 - Property (CSL): $P_{<0.5}(\text{true} \sqcup^{\leq 5} q_1 = 2 \wedge q_2 = 2)$

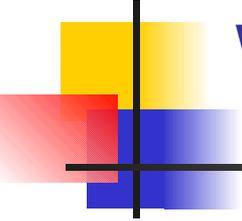


Statistical Solution Method

[Younes & Simmons, CAV-02]

- Use **discrete event simulation** to generate sample paths
- Use **acceptance sampling** to verify probabilistic properties
 - Hypothesis: $P_{\geq \theta}(\rho)$
 - Observation: verify ρ over a sample path

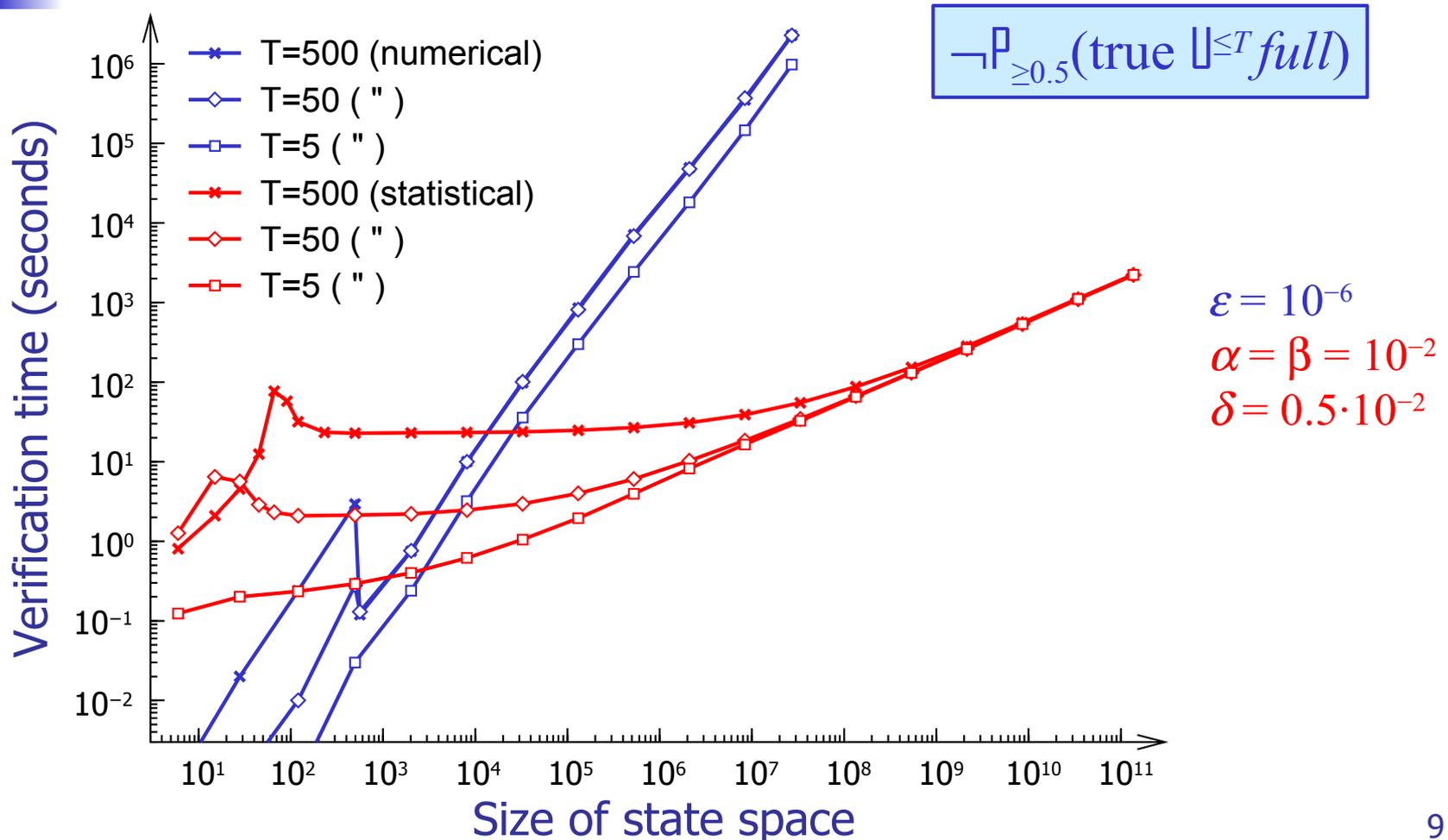
Not estimation!



Why Statistical Approach?

- Benefits
 - Insensitive to size of system
 - Easy to trade accuracy for speed
 - Easy to parallelize
- Alternative: Numerical approach
 - Memory intensive
 - Limited to certain classes of systems

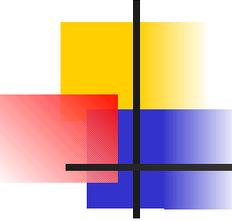
Tandem Queuing Network: Results [Younes et al., TACAS-04]



Tandem Queuing Network: Distributed Sampling

- Use multiple machines to generate observations
 - m_1 : Pentium IV 3GHz
 - m_2 : Pentium III 733MHz
 - m_3 : Pentium III 500MHz

n	% observations				% observations			m_1 only
	m_1	m_2	m_3	time	m_1	m_2	time	time
63	70	20	10	0.46	71	29	0.50	0.58
2047	60	26	14	1.28	70	30	1.46	1.93
65535	65	21	14	26.29	67	33	33.89	44.85

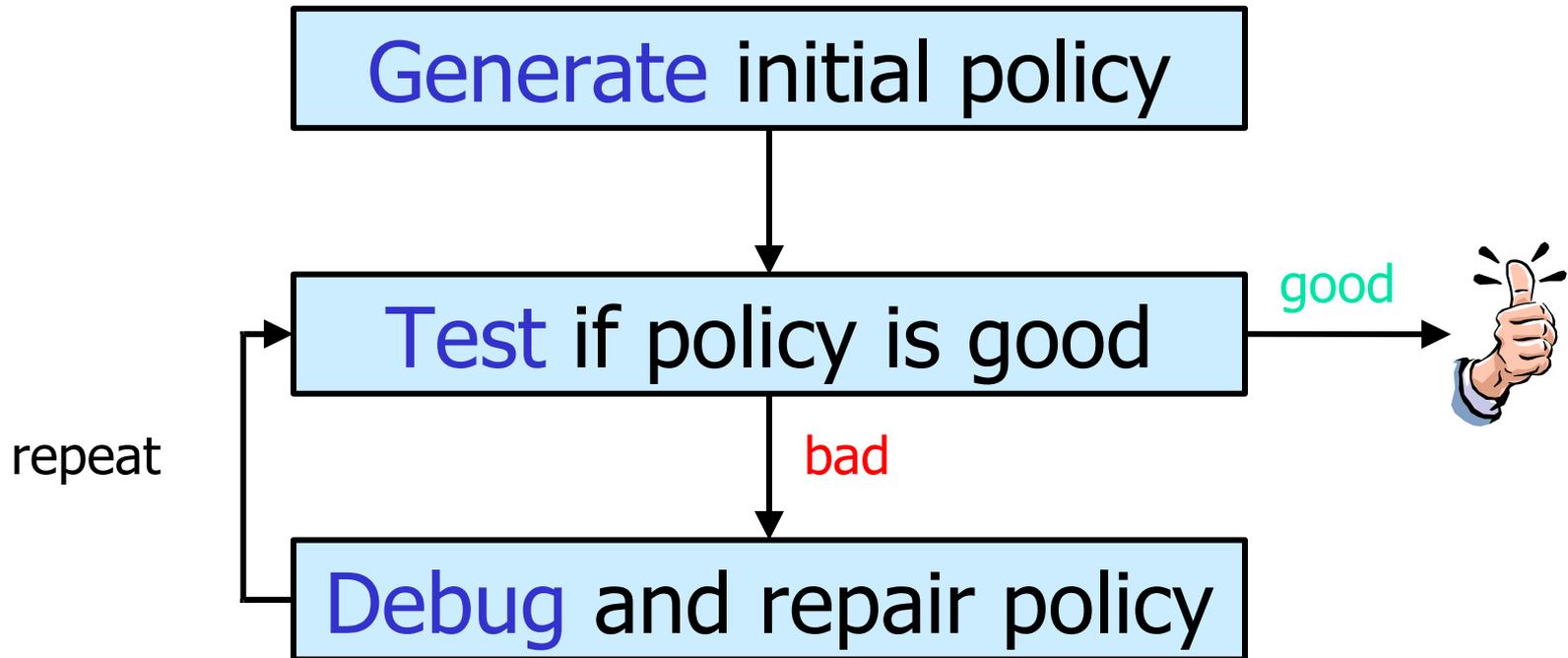


Planning with Asynchronous Events

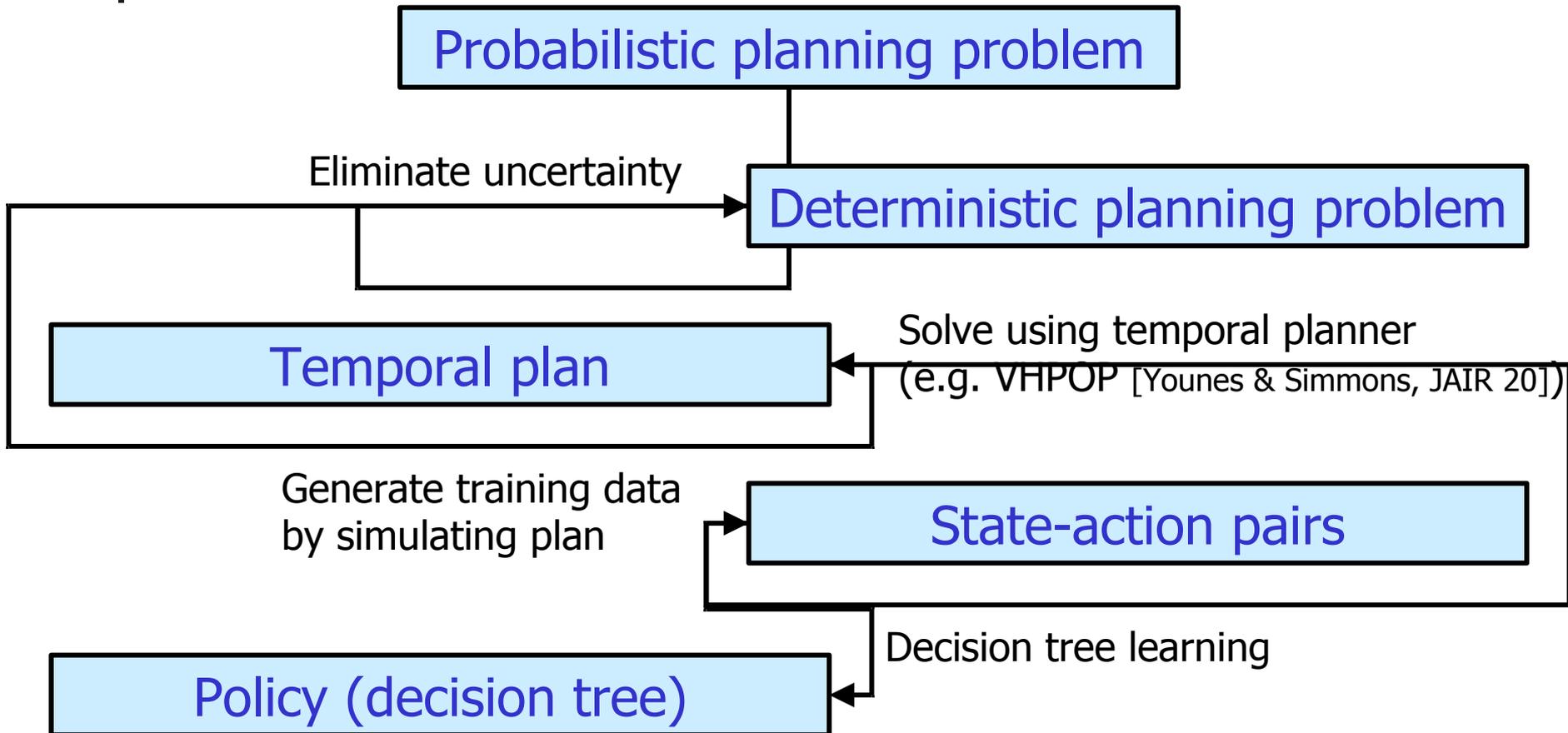
- Goal oriented planning
 - Satisfy goal condition, expressed in CSL
 - Local search with policy repair
[Younes & Simmons, ICAPS-03, ICAPS-04]
- Decision theoretic planning
 - Maximize reward
 - **Generalized** semi-Markov decision process (GSMDP) [Younes & Simmons, AAAI-04]

Generate, Test and Debug

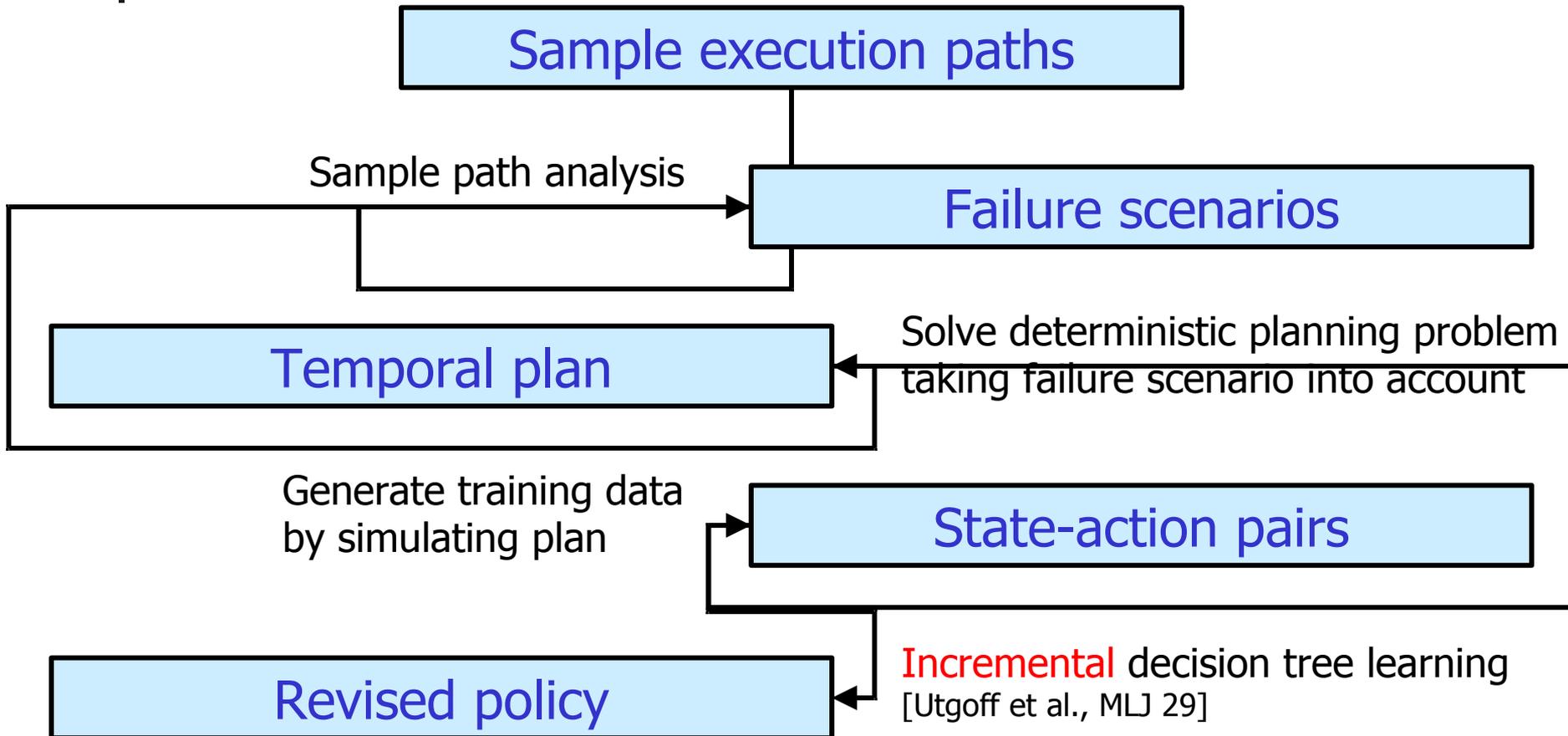
[Simmons, AAAI-88]

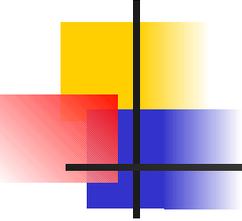


Policy Generation



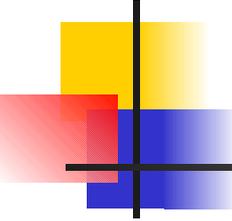
Policy Debugging





A Model of Stochastic Discrete Event Systems

- Generalized semi-Markov process (GSMP) [Matthes 1962]
 - A set of events E
 - A set of states S
- GSMDP
 - Actions $A \subset E$ are controllable events
 - Policy: mapping from execution histories to sets of actions



Events

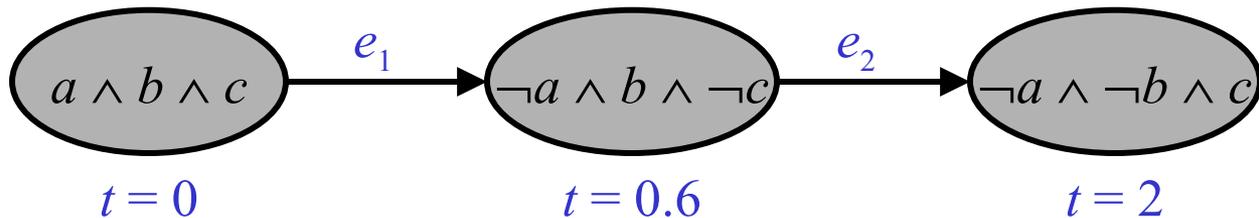
- With each event e is associated:
 - A condition ϕ_e identifying the set of states in which e is **enabled**
 - A distribution G_e governing the time e must remain enabled before it **triggers**
 - A distribution $p_e(s'|s)$ determining the probability that the next state is s' if e triggers in state s

Events: Example

- Three events: e_1, e_2, e_3

- $\phi_1 = a, \phi_2 = b, \phi_3 = c$

- $G_1 = \text{Exp}(1), G_2 = D(2), G_3 = U(0,1)$



$$G_1 = \text{Exp}(1)$$

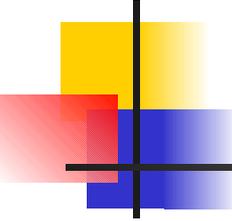
$$G_2 = D(2)$$

$$G_3 = U(0,1)$$

$$G_2 = D(1.4)$$

$$G_3 = U(0,1)$$

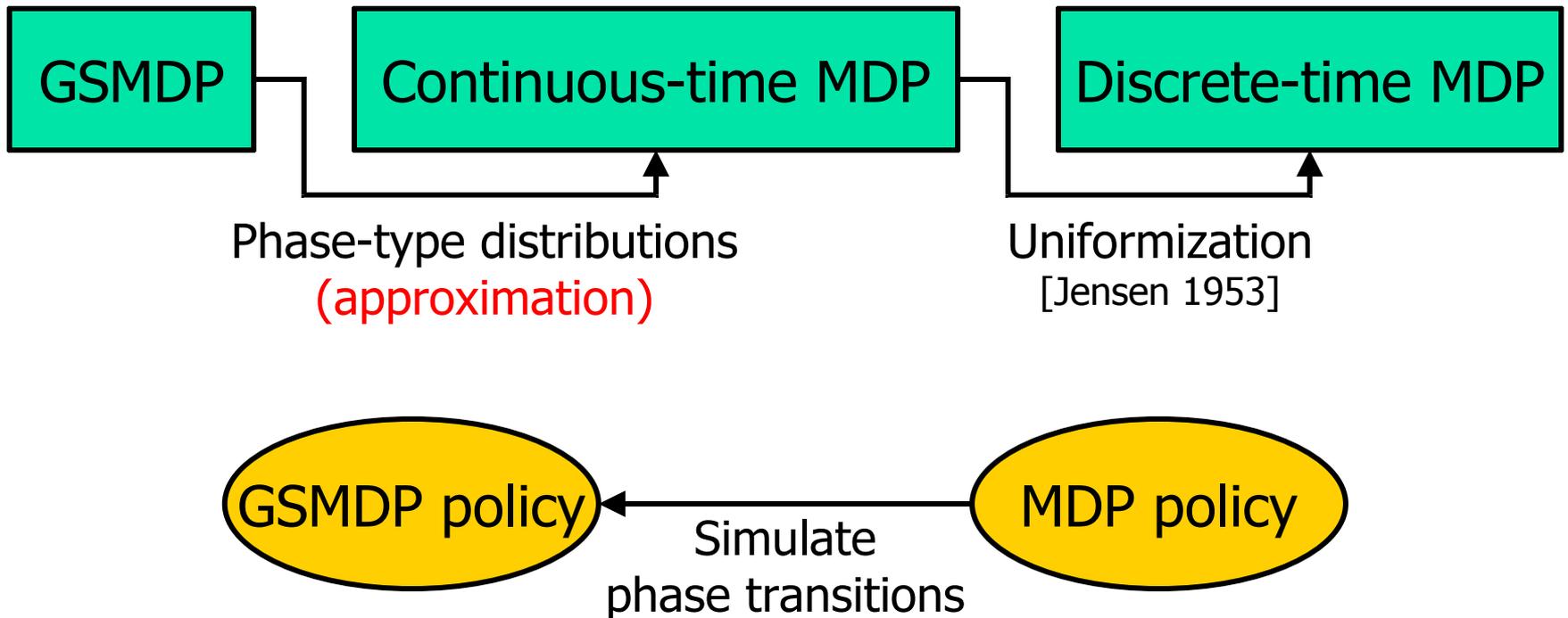
Asynchronous events \Rightarrow beyond semi-Markov



Rewards and Optimality

- Lump sum reward $k(s, e, s')$ associated with transition from s to s' caused by e
- Continuous reward rate $r(s, A)$ associated with A being enabled in s
- Infinite-horizon discounted reward
 - Unit reward earned at time t counts as $e^{-\alpha t}$
- Optimal choice may depend on entire execution history

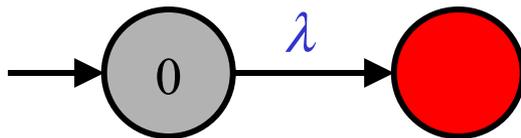
GSMDP Solution Method



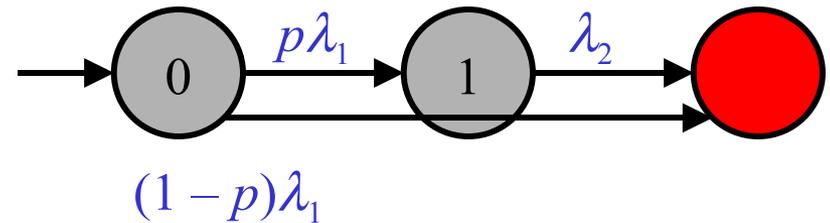
Continuous Phase-Type Distributions [Neuts 1981]

- Time to absorption in a continuous-time Markov chain with n transient states

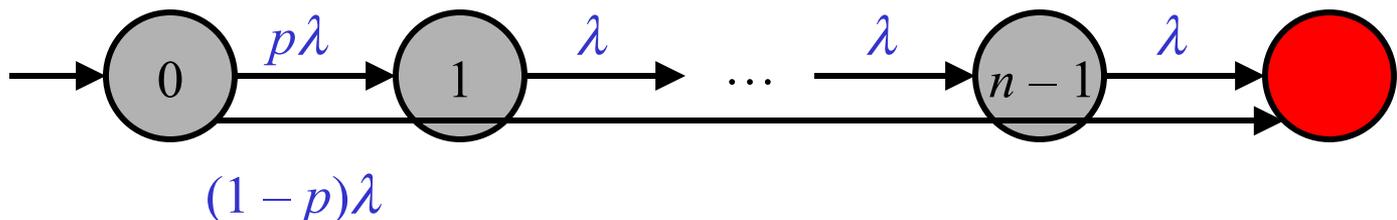
Exponential

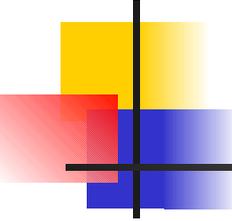


Two-phase Coxian



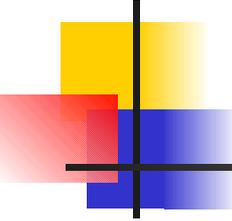
n -phase generalized Erlang





Method of Moments

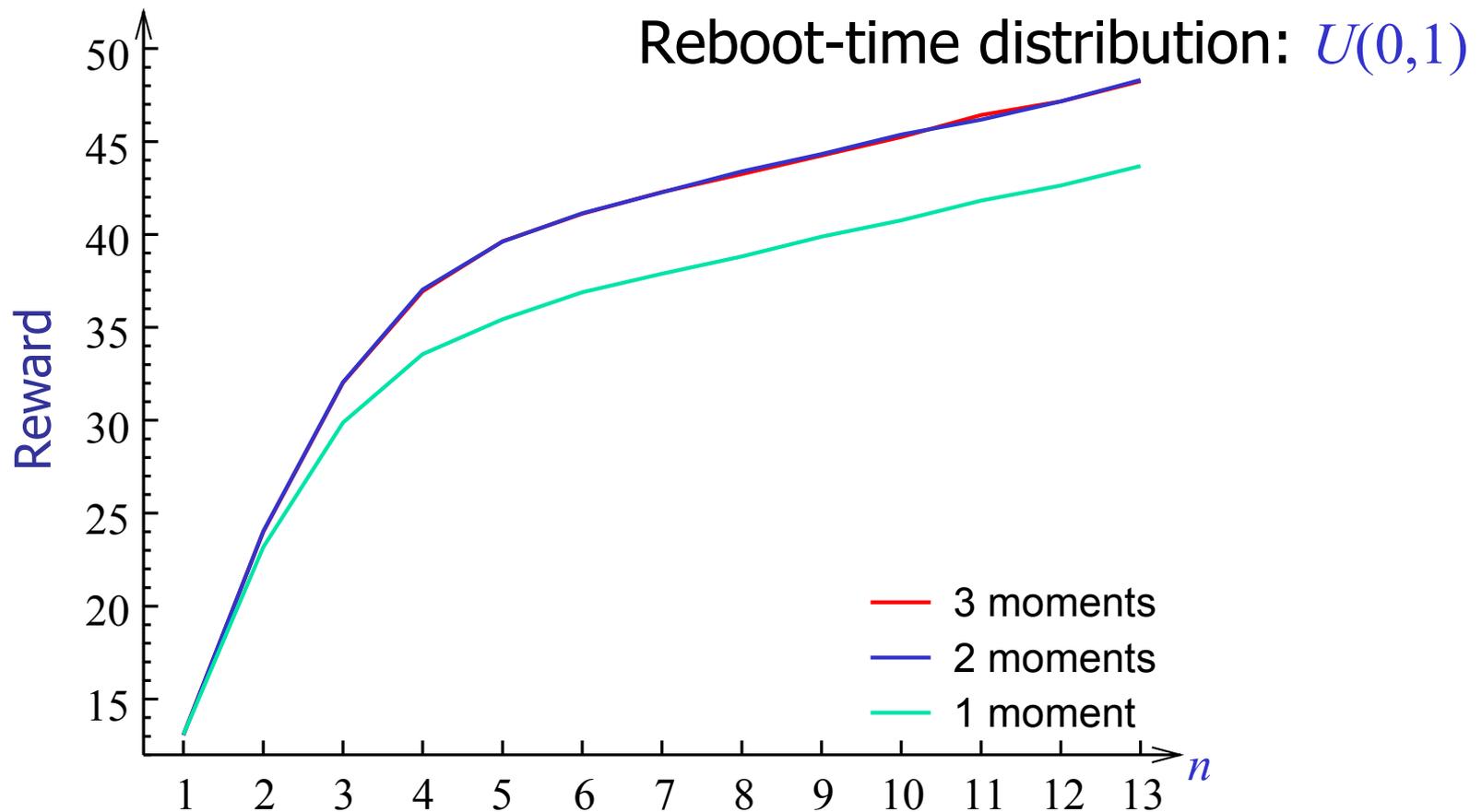
- **Approximate** general distribution G with phase-type distribution PH by matching the first k moments
 - Mean (first moment): μ_1
 - Variance: $\sigma^2 = \mu_2 - \mu_1^2$
 - The i th moment: $\mu_i = E[X^i]$



System Administration

- Network of n machines
- Reward rate $c(s) = k$ in states where k machines are up
- One crash event and one reboot action per machine
 - At most one action enabled at any time (single agent)

System Administration: Performance



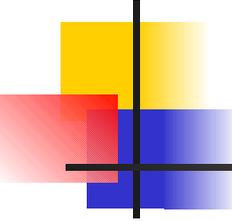
System Administration: Performance

size	1 moment		2 moments		3 moments	
	states	time (s)	states	time (s)	states	time (s)
4	16	0.36	32	3.57	112	10.30
5	32	0.82	80	7.72	272	22.33
6	64	1.89	192	16.24	640	40.98
7	128	3.65	448	28.04	1472	69.06
8	256	6.98	1024	48.11	3328	114.63
9	512	16.04	2304	80.27	7424	176.93
10	1024	33.58	5120	136.4	16384	291.70
11	2048	66.00	24576	264.17	35840	481.10
12	4096	111.96	53248	646.97	77824	1051.33
13	8192	210.03	114688	2588.95	167936	3238.16

$$2^n$$

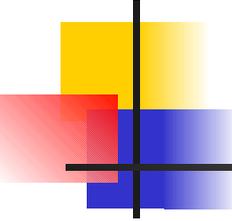
$$(n+1)2^n$$

$$(1.5n+1)2^n$$



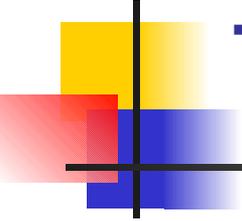
Summary

- Asynchronous processes exist in the real world
- The complexity is manageable
 - Model checking: statistical hypothesis testing and simulation
 - Planning: sample path analysis and phase-type distributions



Challenges Ahead

- Evaluation of policy repair techniques
- Complexity results for optimal GSMDP planning (undecidable?)
- Discrete phase-type distributions
 - Handles deterministic distributions



Tools

- Ymer
 - Statistical probabilistic model checking
- Tempastic-DTP
 - Decision theoretic planning with asynchronous events