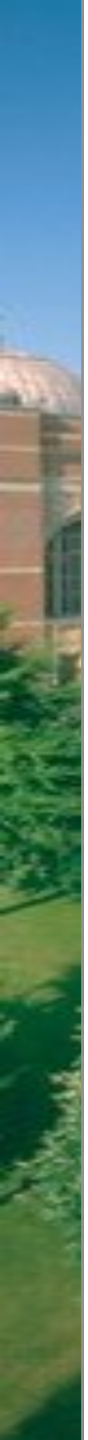


Probabilistic Model Checking and Strategy Synthesis

Dave Parker

University of Birmingham

NASA Ames, October 2013

Probabilistic Model Checking and Strategy Synthesis

Dave Parker

University of Birmingham

Joint work with: Marta Kwiatkowska, Vojtěch Forejt, Gethin Norman, Hongyang Qu, Aistis Simaitis, Taolue Chen, …

Overview

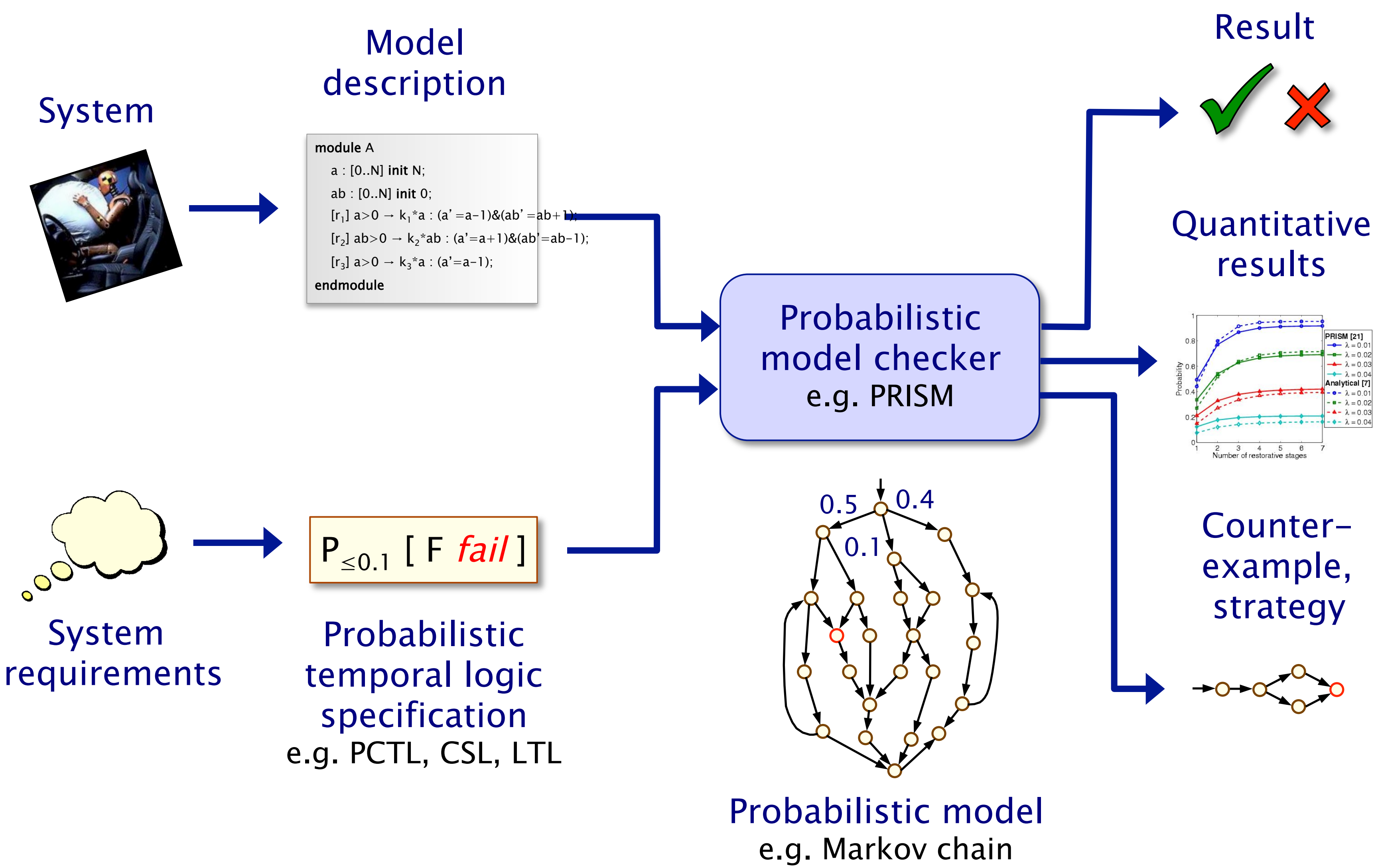
- Probabilistic model checking & PRISM
  - example: Bluetooth
- Verification vs. strategy synthesis
  - Markov decision processes (MDPs)
  - example: robot controller
- Multi-objective probabilistic model checking
  - examples: team-formation/power management/…
- Model checking stochastic games
  - example: energy management

Motivation

- Verifying probabilistic systems…
  - unreliable or unpredictable behaviour
    - failures of physical components
    - message loss in wireless communication
    - unreliable sensors/actuators
  - randomisation in algorithms/protocols
    - random back-off in communication protocols
    - random routing to reduce flooding or provide anonymity

- We need to verify quantitative system properties
  - "the probability of the airbag failing to deploy within 0.02 seconds of being triggered is at most 0.001"
  - not just correctness: reliability, timeliness, performance, …

Probabilistic model checking



Probabilistic model checking

- Various types of probabilistic models supported

PRISM models

- discrete-time Markov chains (DTMCs)
- continuous-time Markov chains (CTMCs)
- Markov decision processes (MDPs)
- probabilistic automata (PAs)
- probabilistic timed automata (PTAs)
- stochastic multi-player games (SMGs)

Probabilistic model checking

- Various types of probabilistic models supported
- Wide range of quantitative properties, expressible in temporal logics (probabilities, timing, costs, rewards, …)

Example PRISM properties

PCTL (reachability) → $P_{\leq 0.1}$ [F *fail*] – "the probability of a failure is at most 0.1"

CSL → $S_{>0.999}$ [*up*] – "the long-run probability of availability is >0.999"

costs & rewards → $R_{\{time\}<100}$ [F *done*] – "the expected termination time is at most 100 seconds"

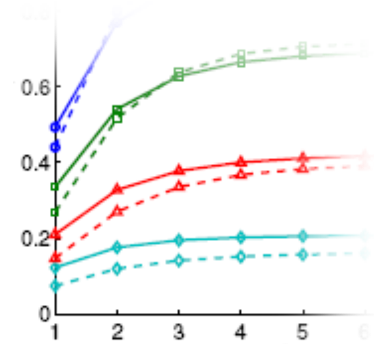
probabilistic LTL → $P_{\geq 0.75}$ [(G $\neg$*hazard*) $\wedge$ (GF *goal*)] – "the probability of avoiding the hazard visiting the goal infinitely often is ≥0.75"

Probabilistic model checking

- Various types of probabilistic models supported
- Wide range of quantitative properties, expressible in temporal logics (probabilities, timing, costs, rewards, …)
- Often focus on numerical results (probabilities etc.)
  - analyse trends, look for system flaws, anomalies

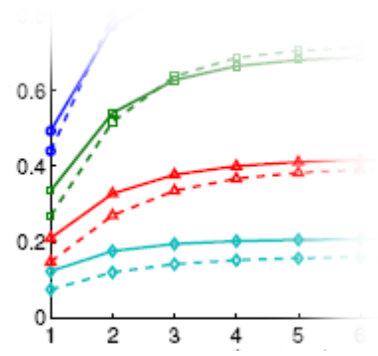
- $P_{\leq 0.1}$ [F *fail*] – "the probability of a failure occurring is at most 0.1"

- $P_{=?}$ [F *fail*] – "what is the probability of a failure occurring?"



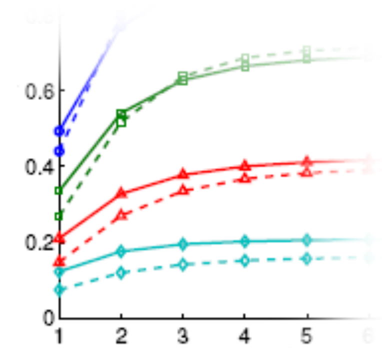
Probabilistic model checking

- Various types of probabilistic models supported
- Wide range of quantitative properties, expressible in temporal logics (probabilities, timing, costs, rewards, …)
- Often focus on numerical results (probabilities etc.)
  - analyse trends, look for system flaws, anomalies
- Provides "exact" numerical results
  - compared to, for example, simulation



Probabilistic model checking

- Various types of probabilistic models supported
- Wide range of quantitative properties, expressible in temporal logics (probabilities, timing, costs, rewards, …)
- Often focus on numerical results (probabilities etc.)
  - analyse trends, look for system flaws, anomalies
- Provides "exact" numerical results
  - compared to, for example, simulation
- Combines numerical & exhaustive analysis
  - especially useful for nondeterministic models

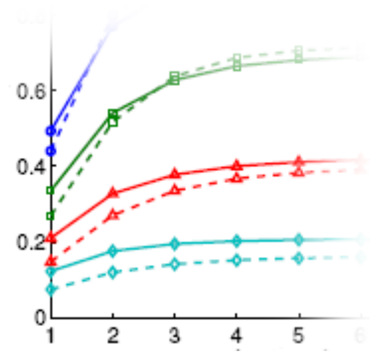


- $P_{=?}$ [F *fail* {*trigger*}{max}]

- $P_{max=?}$ [F *fail*]

Probabilistic model checking

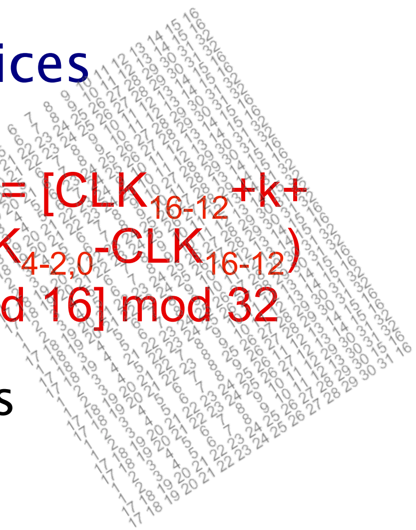
- Various types of probabilistic models supported
- Wide range of quantitative properties, expressible in temporal logics (probabilities, timing, costs, rewards, …)
- Often focus on numerical results (probabilities etc.)
  - analyse trends, look for system flaws, anomalies
- Provides "exact" numerical results
  - compared to, for example, simulation
- Combines numerical & exhaustive analysis
  - especially useful for nondeterministic models



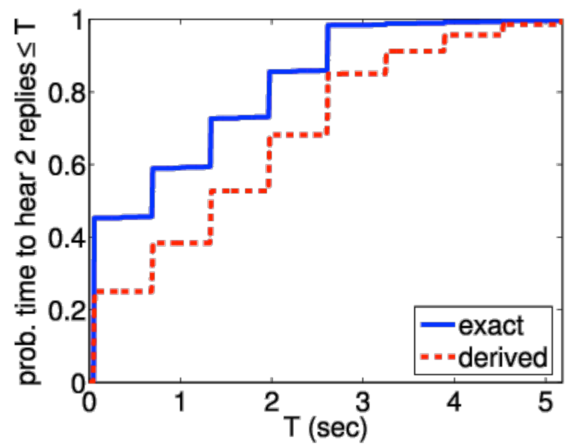
- Flexible, fully automated & widely applicable
  - network/communication protocols, security, robotics & planning, power management, nanotechnology, biology…

Case study: Bluetooth

- Device discovery between pair of Bluetooth devices
  - performance essential for this phase
- Complex discovery process
  - two asynchronous 28-bit clocks
  - pseudo-random hopping between 32 frequencies
  - random waiting scheme to avoid collisions
  - 17,179,869,184 initial configurations (too many to sample effectively)
- Probabilistic model checking (PRISM)
  - e.g. “worst-case expected discovery time is at most 5.17s”
  - e.g. “probability discovery time exceeds 6s is always < 0.001”
  - shows weaknesses in simplistic analysis



freq = [$CLK_{16\text{-}12}$+k+ ($CLK_{4\text{-}2,0}$-$CLK_{16\text{-}12}$) mod 16] mod 32

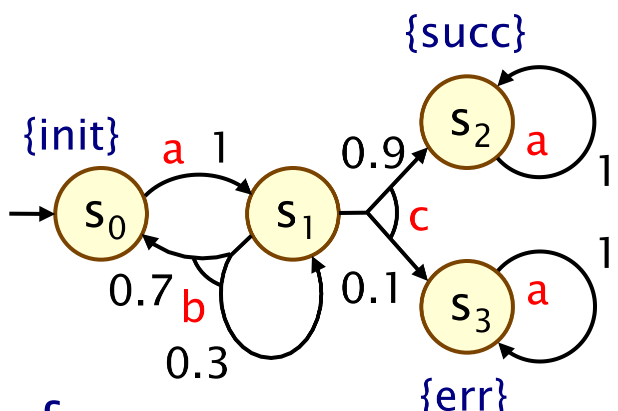


Overview

- Probabilistic model checking & PRISM
  - example: Bluetooth
- Verification vs. strategy synthesis
  - Markov decision processes (MDPs)
  - example: robot controller
- Multi-objective probabilistic model checking
  - examples: team-formation/power management/…
- Model checking stochastic games
  - example: energy management

Markov decision processes (MDPs)

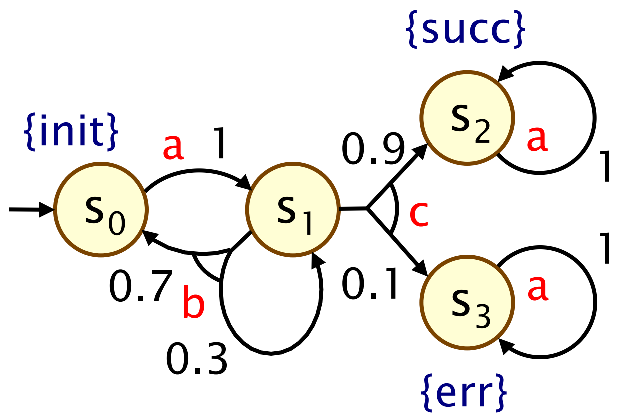
- Markov decision processes (MDPs)
  - model nondeterministic as well as probabilistic behaviour
  - widely used also in planning, optimal control, …
  - nondeterministic choice between probability distributions



- Nondeterminism for:
  - concurrency/scheduling: interleavings of parallel components
  - abstraction, or under-specification, of unknown behaviour
  - adversarial behaviour of the environment, or control

Strategies

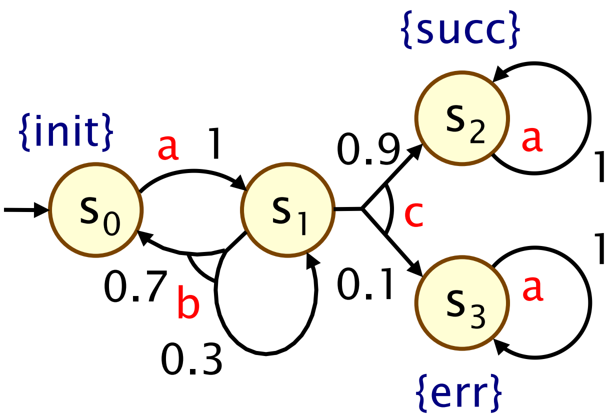
- A strategy (or "policy" or "adversary")
  - is a resolution of nondeterminism, based on history
  - is (formally) a mapping σ from finite paths to distributions
  - induces an (infinite-state) discrete-time Markov chain



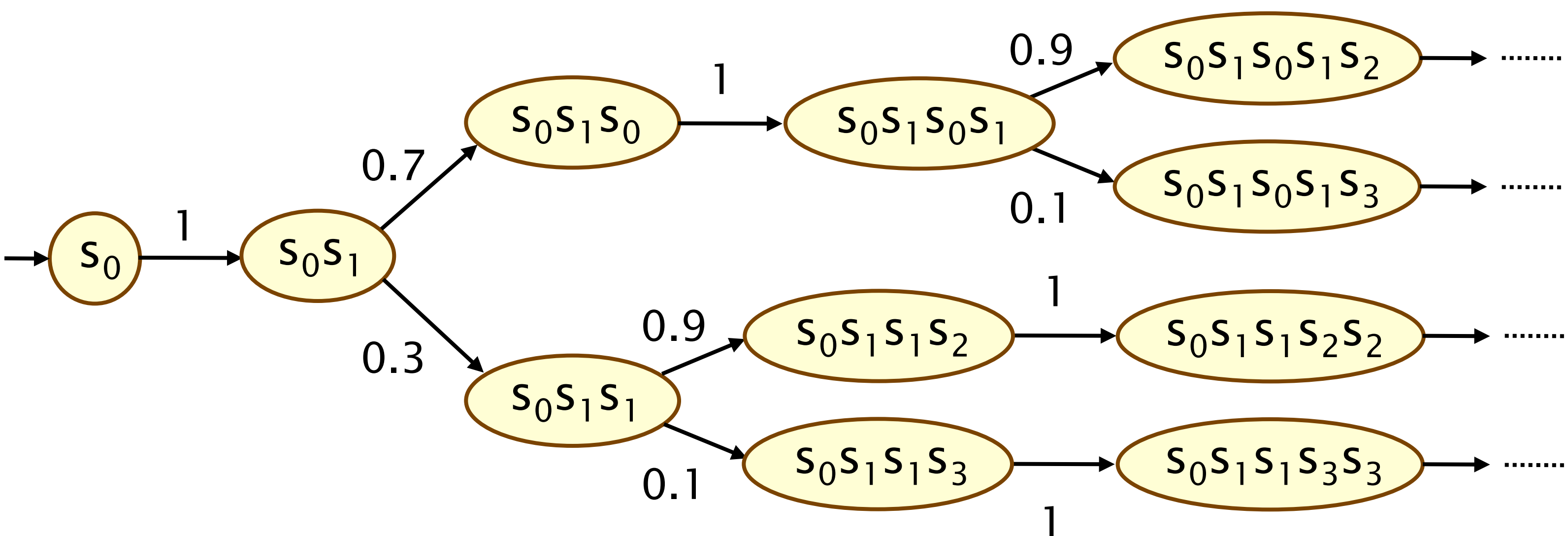
- Classes of strategies:
  - randomisation: deterministic or randomised
  - memory: memoryless, finite-memory, or infinite-memory

Example strategy

- Strategy σ which picks b then c in s_1
  - σ is finite-memory and deterministic

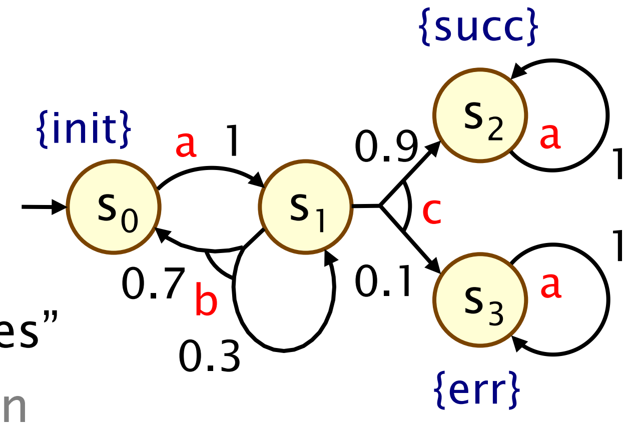


- Fragment of induced Markov chain:



Verification vs. Strategy synthesis

- 1. Verification
  - quantify over all possible strategies (i.e. best/worst-case)
  - $P_{\leq 0.01}$ [F *err*] : “the probability of an error occurring is $\leq$ 0.01 for all strategies”
  - applications: randomised communication protocols, randomised distributed algorithms, security, …

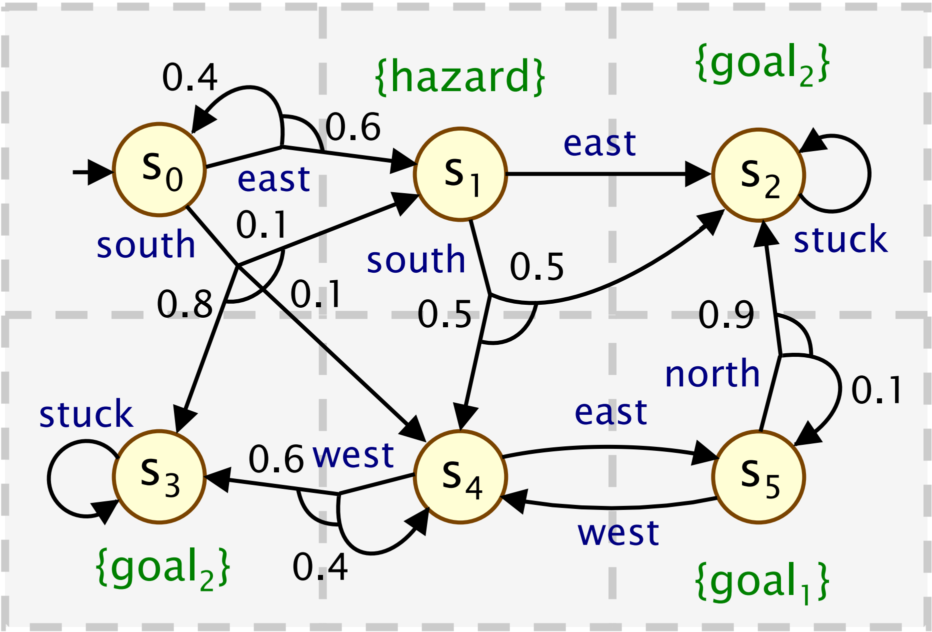


- 2. Strategy synthesis
  - generation of "correct-by-construction" controllers
  - $P_{\leq 0.01}$ [F *err*] : "does there exist a strategy for which the probability of an error occurring is $\leq$ 0.01?”
  - applications: robotics, power management, security, …

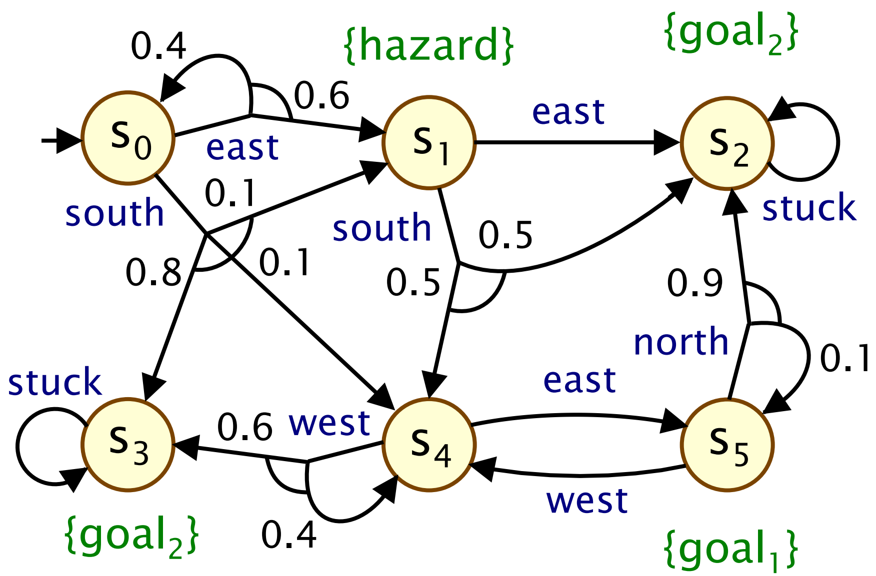
- Two dual problems; same underlying computation:
  - compute optimal (minimum or maximum) values

Running example

- Example MDP
  - robot moving through terrain divided in to 3 x 2 grid



Example – Reachability



Verify: $P_{\leq 0.6}$ [F $goal_1$]

or

Synthesise for: $P_{\geq 0.4}$ [F $goal_1$]

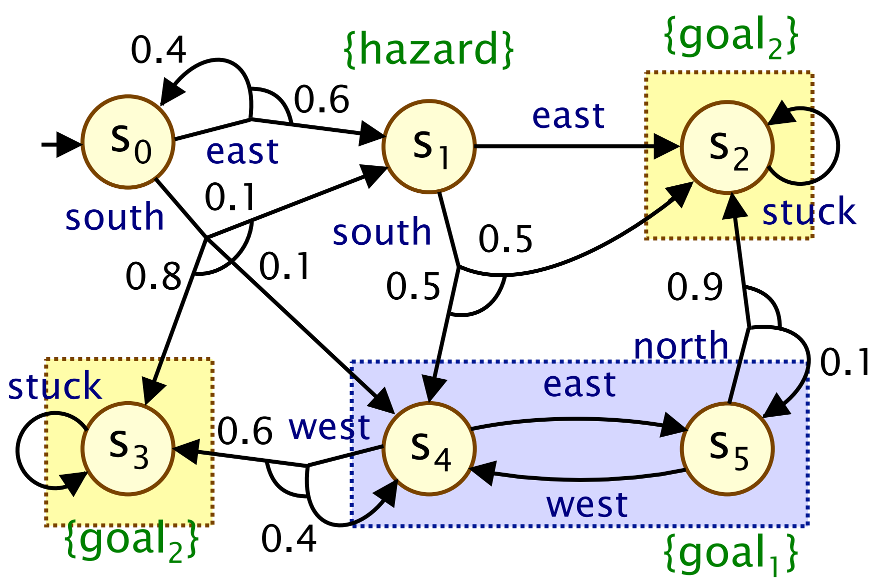
$\Downarrow$

Compute: $P_{max=?}$ [F $goal_1$]

Optimal strategies:
memoryless and deterministic

Computation: graph analysis
& linear programming problem

Example – Reachability



Verify: $P_{\leq 0.6}$ [F goal$_1$]

or

Synthesise for: $P_{\geq 0.4}$ [F goal$_1$]

$\Downarrow$

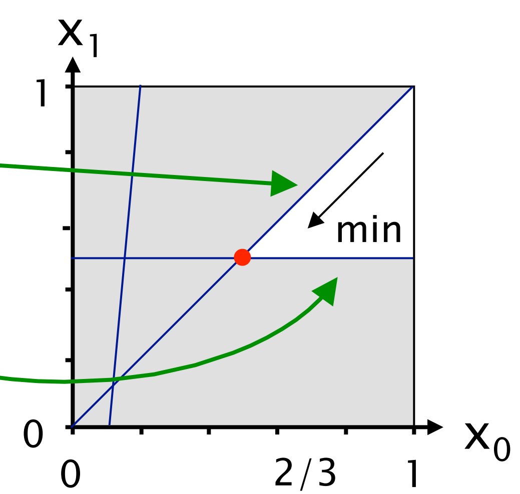
Compute: $P_{max=?}$ [F goal$_1$] = 0.5

Optimal strategies:
memoryless and deterministic

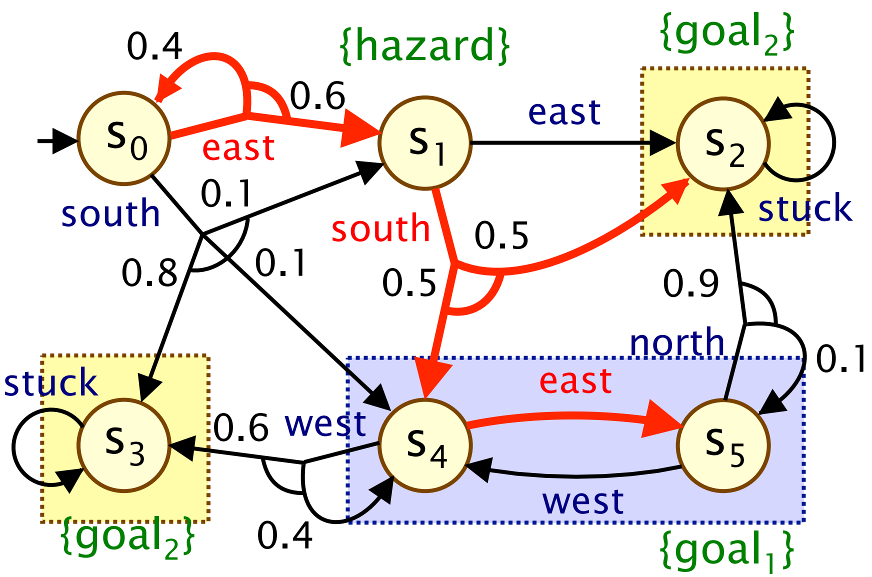
Computation: graph analysis
& linear programming problem

$x_0 \geq x_1$
(east)

$x_1 \geq 0.5$
(south)



Example – Reachability



Verify: $P_{\leq 0.6}$ [F goal$_1$]

or

Synthesise for: $P_{\geq 0.4}$ [F goal$_1$]

$\Downarrow$

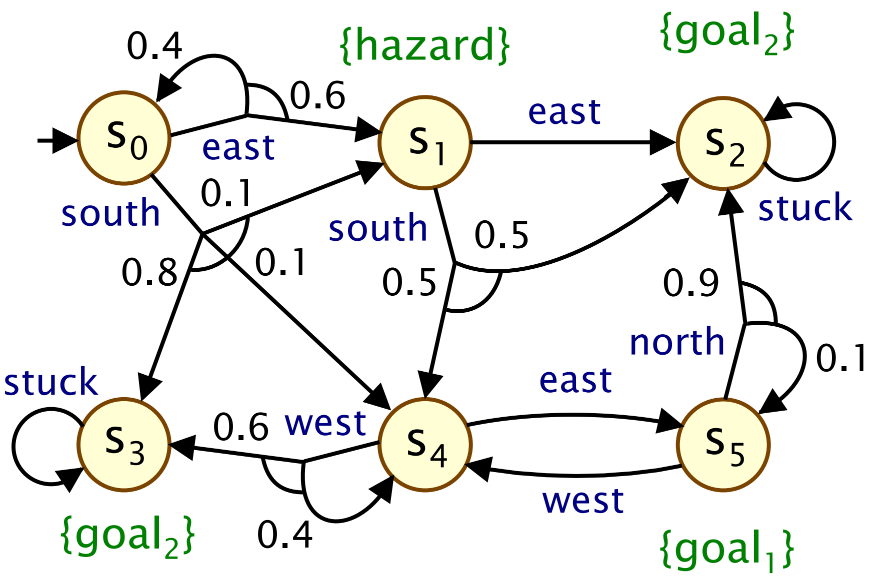
Compute: $P_{max=?}$ [F goal$_1$] = 0.5

Optimal strategies:
memoryless and deterministic

Computation: graph analysis
& linear programming problem

Optimal strategy:
s_0 : east
s_1 : south
s_2 : –
s_3 : –
s_4 : east
s_5 : –

Example - Costs/rewards



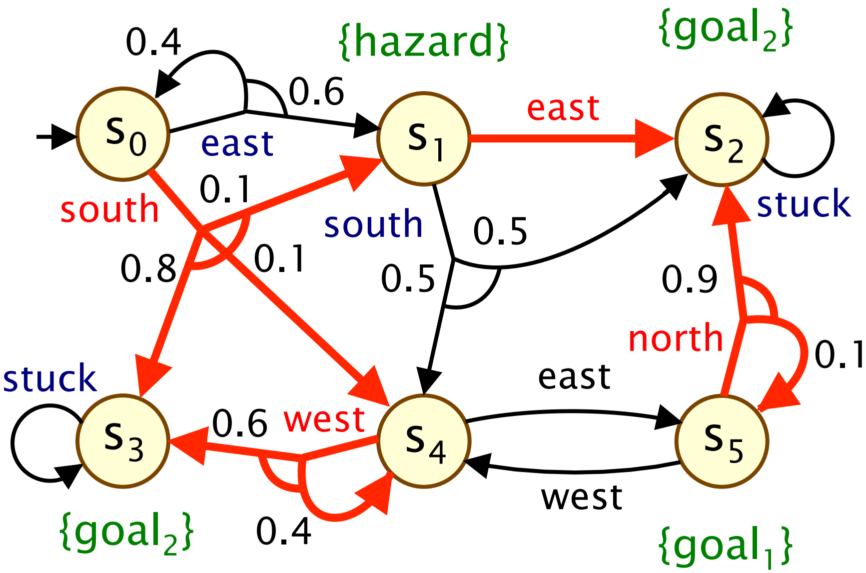
$R_{min=?}$ [F goal$_2$]

"what is the minimum expected number of moves needed to reach goal$_2$?"

Optimal strategies:
memoryless and deterministic

Computation: graph analysis & linear programming problem

Example - Costs/rewards



$R_{min=?}$ [F goal$_2$] = 19/15

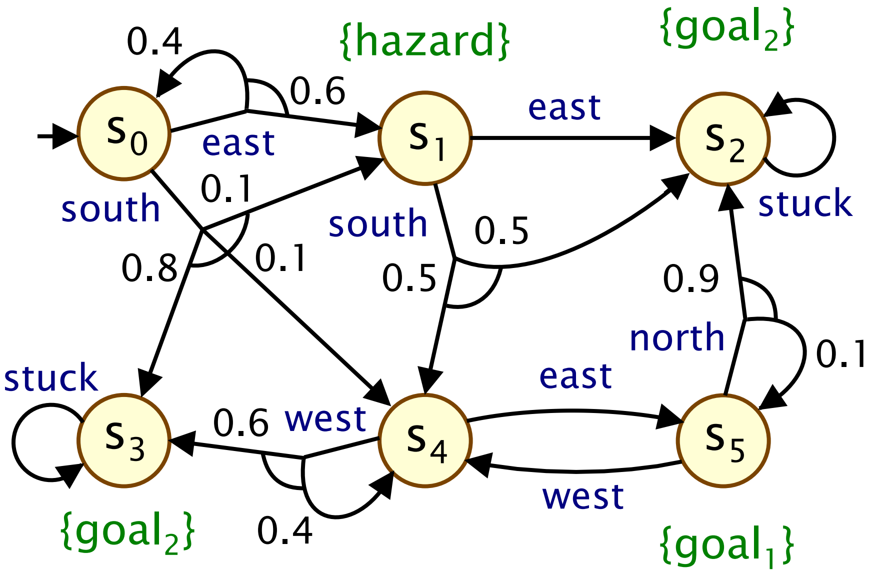
"what is the minimum expected number of moves needed to reach goal$_2$?"

Optimal strategies:
memoryless and deterministic

Computation: graph analysis & linear programming problem

Optimal strategy:
s_0 : south
s_1 : east
s_2 : –
s_3 : –
s_4 : west
s_5 : north

Example – LTL



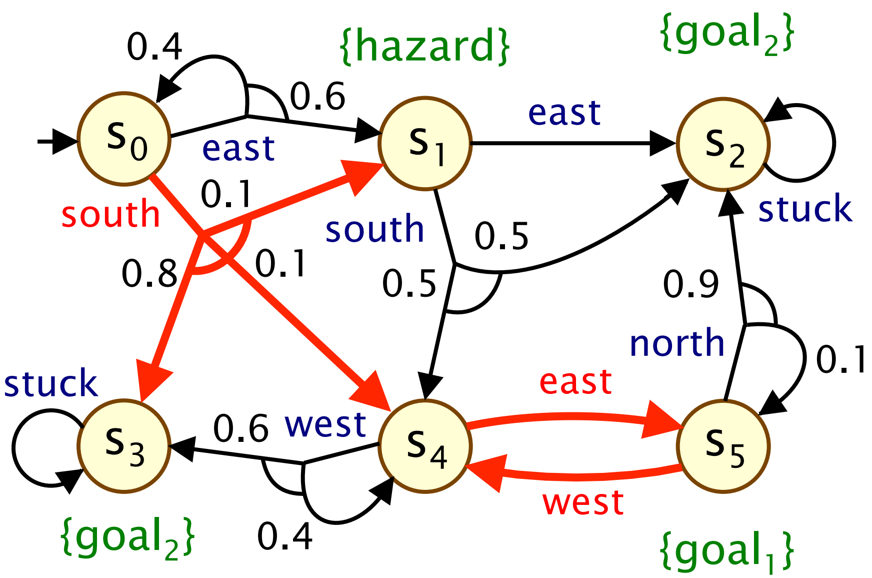
$P_{max=?}$ [(G¬hazard) ∧ (GF goal$_1$)]

"what is the maximum probability of avoiding hazard and visiting goal$_1$ infinitely often?"

Optimal strategies:
finite-memory and deterministic

Computation:
construct product of MDP and a deterministic ω-automaton;
then probabilistic reachability

Example - LTL



$P_{max=?}$ [(G¬hazard) ∧ (GF goal$_1$)]

"what is the maximum probability of avoiding hazard and visiting goal$_1$ infinitely often?" = 0.1

Optimal strategies:
finite-memory and deterministic

Computation:
construct product of MDP and a deterministic ω-automaton; then probabilistic reachability

Optimal strategy:
s_0 : south
s_1 : -
s_2 : -
s_3 : -
s_4 : east
s_5 : west

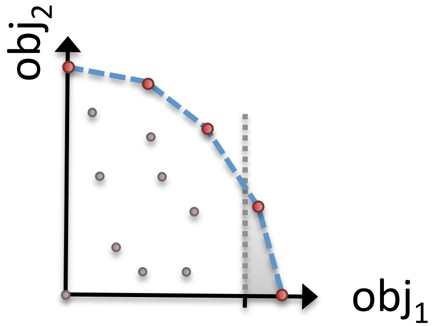
In this instance, memoryless (not usually)

Overview

- Probabilistic model checking & PRISM
  - example: Bluetooth
- Verification vs. strategy synthesis
  - Markov decision processes (MDPs)
  - example: robot controller
- Multi-objective probabilistic model checking
  - examples: team-formation/power management/…
- Model checking stochastic games
  - example: energy management

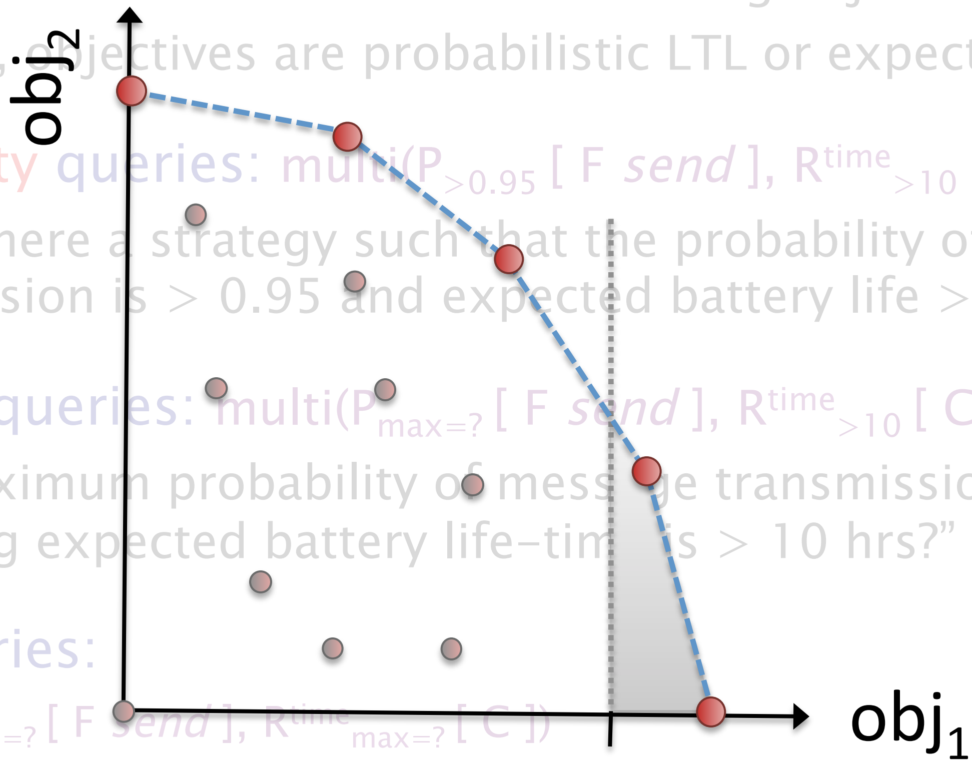
Multi-objective model checking

- Multi-objective probabilistic model checking
  - investigate trade-offs between conflicting objectives
  - in PRISM, objectives are probabilistic LTL or expected rewards
- Achievability queries: multi($P_{>0.95}$ [F *send*], $R^{time}_{>10}$ [C])
  - e.g. "is there a strategy such that the probability of message transmission is > 0.95 and expected battery life > 10 hrs?"
- Numerical queries: multi($P_{max=?}$ [F *send*], $R^{time}_{>10}$ [C])
  - e.g. "maximum probability of message transmission, assuming expected battery life-time is > 10 hrs?"
- Pareto queries:
  - multi($P_{max=?}$ [F *send*], $R^{time}_{max=?}$ [C])
  - e.g. "Pareto curve for maximising probability of transmission and expected battery life-time"



Multi-objective model checking

- Multi-objective probabilistic model checking
  - investigate trade-offs between conflicting objectives
  - in PRISM, objectives are probabilistic LTL or expected rewards
- Achievability queries: multi($P_{>0.95}$ [F *send*], $R^{time}_{>10}$ [C])
  - e.g. "is there a strategy such that the probability of message transmission is > 0.95 and expected battery life > 10 hrs?"
- Numerical queries: multi($P_{max=?}$ [F *send*], $R^{time}_{>10}$ [C])
  - e.g. "maximum probability of message transmission, assuming expected battery life-time is > 10 hrs?"
- Pareto queries:
  - multi($P_{max=?}$ [F *send*], $R^{time}_{max=?}$ [C])
  - e.g. "Pareto curve for maximising probability of transmission and expected battery life-time"



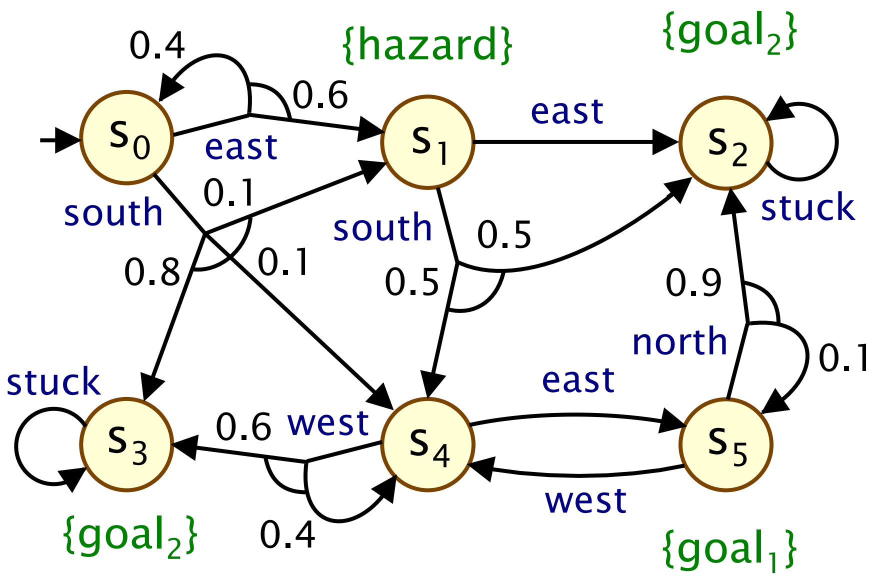
Multi-objective model checking

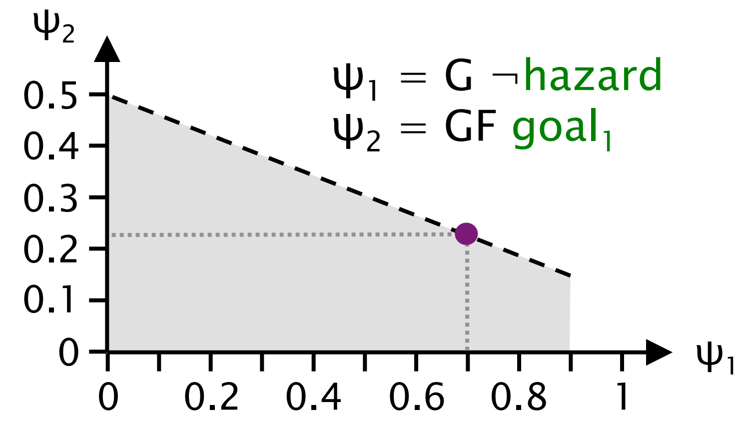
- Optimal strategies for multiple objectives
  - may be randomised
  - and finite-memory (when using LTL formulae)

- Multi-objective probabilistic model checking
  - reduces to linear programming,
    on an MDP-automata product [TACAS'07,TACAS'11]
  - can be approximated using iterative numerical methods,
    via approximation of the Pareto curve [ATVA'12]

- Extensions [ATVA'12]
  - arbitrary Boolean combinations of objectives
    - e.g. $\psi_1 \Rightarrow \psi_2$ (all strategies satisfying ψ_1 also satisfy ψ_2)
  - time-bounded (finite-horizon) properties

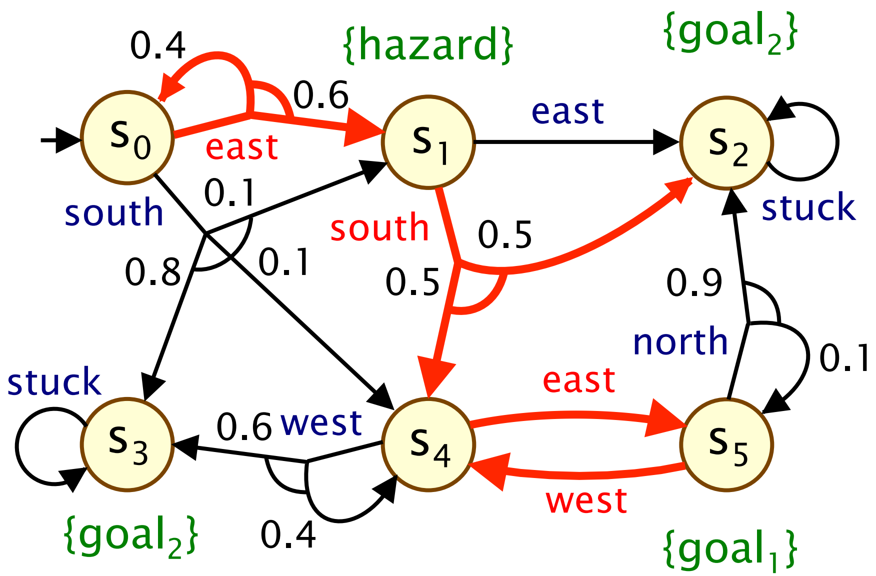
Example – Multi-objective





- Achievability query
  - $P_{\geq 0.7}$ [G ¬hazard] ∧ $P_{\geq 0.2}$ [GF goal$_1$] ? True (achievable)
- Numerical query
  - $P_{max=?}$ [GF goal$_1$] such that $P_{\geq 0.7}$ [G ¬hazard] ? ~0.2278
- Pareto query
  - for $P_{max=?}$ [G ¬hazard] ∧ $P_{max=?}$ [GF goal$_1$] ?

Example - Multi-objective



Strategy 1
(deterministic)

s_0 : east

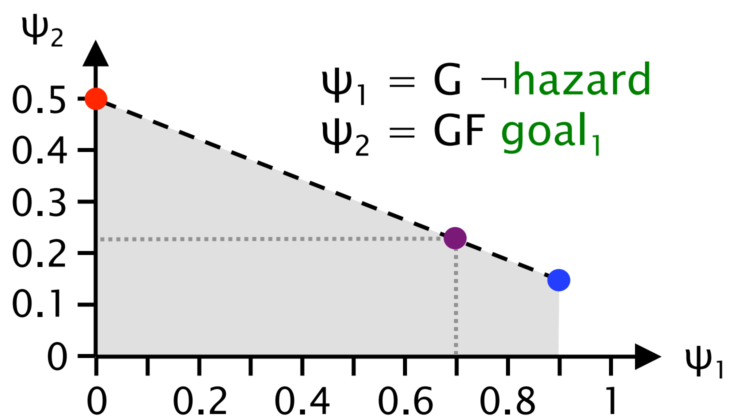
s_1 : south

s_2 : –

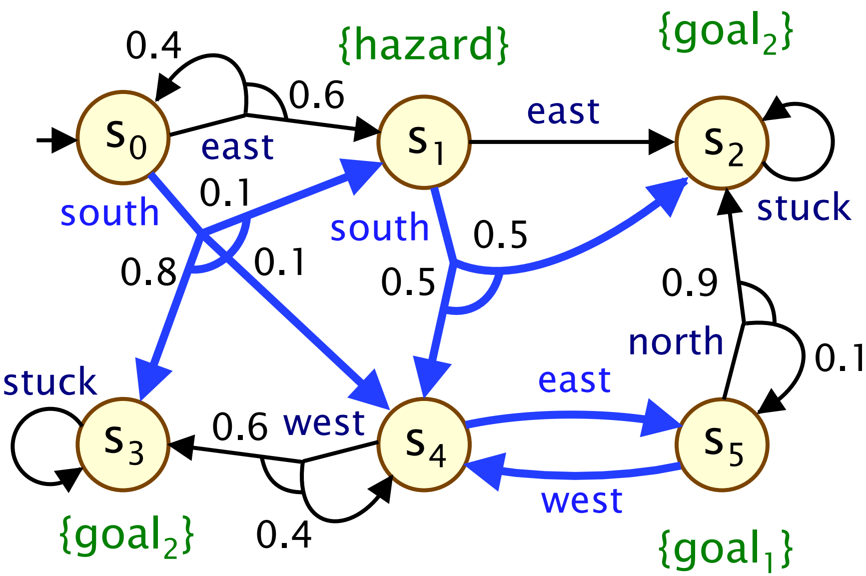
s_3 : –

s_4 : east

s_5 : west



Example – Multi-objective



Strategy 2 (deterministic)

s_0 : south

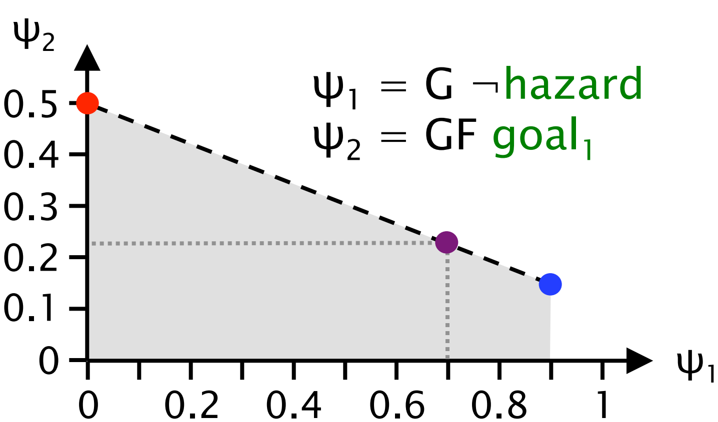
s_1 : south

s_2 : –

s_3 : –

s_4 : east

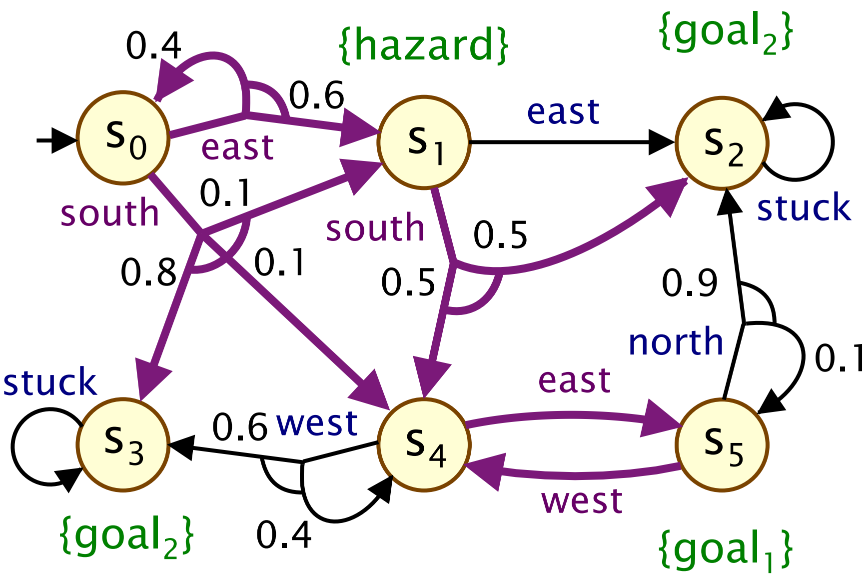
s_5 : west

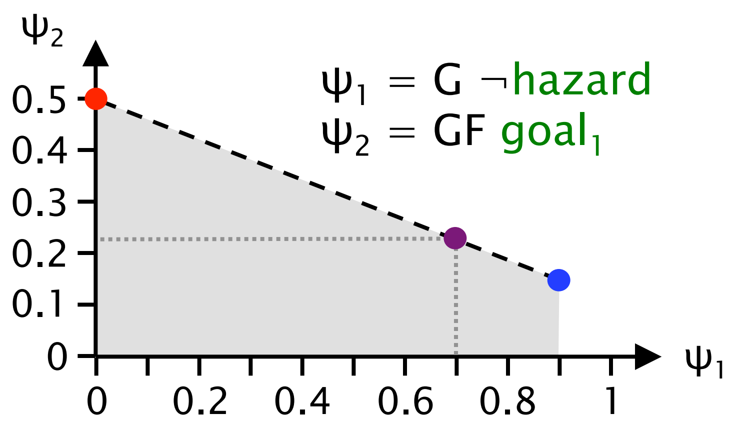


$\psi_1 = G\ \neg hazard$

$\psi_2 = GF\ goal_1$

Example – Multi-objective





Optimal strategy:
(randomised)

s_0 : 0.3226 : east
0.6774 : south

s_1 : 1.0 : south

s_2 : –

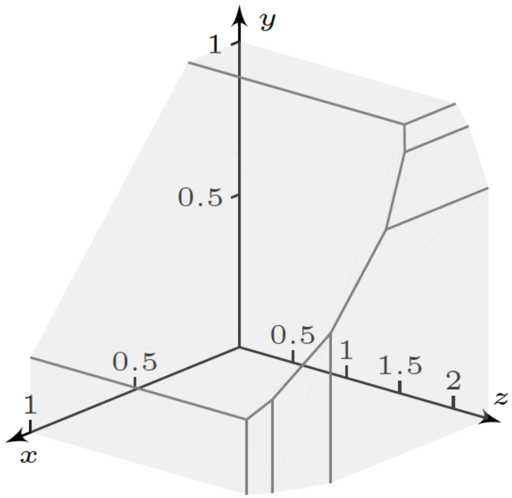
s_3 : –

s_4 : 1.0 : east

s_5 : 1.0 : west

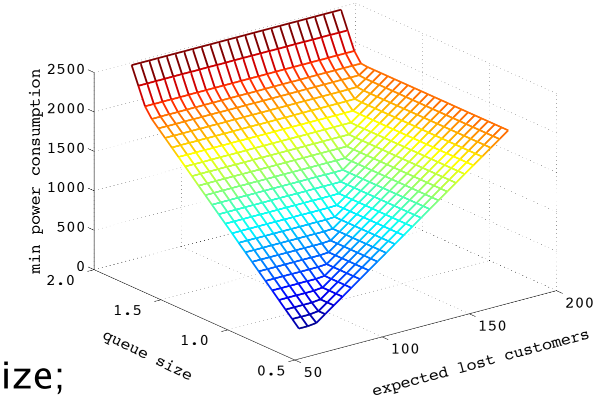
Multi-objective: Applications

Synthesis of team formation strategies [CLIMA'11, ATVA'12]



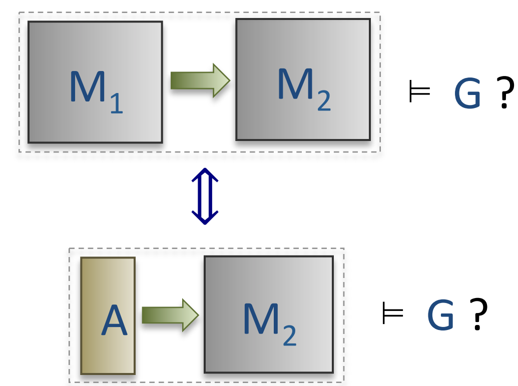
Pareto curve:
x="probability of completing task 1";
y="probability of completing task 2";
z="expected size of successful team"

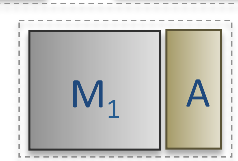
Synthesis of dynamic power management controllers [TACAS'11]



"minimise energy consumption, subject to constraints on:
(i) expected job queue size;
(ii) expected number of lost jobs

Probabilistic assume-guarantee framework [TACAS'10, TACAS'11, Info&Comp'13]





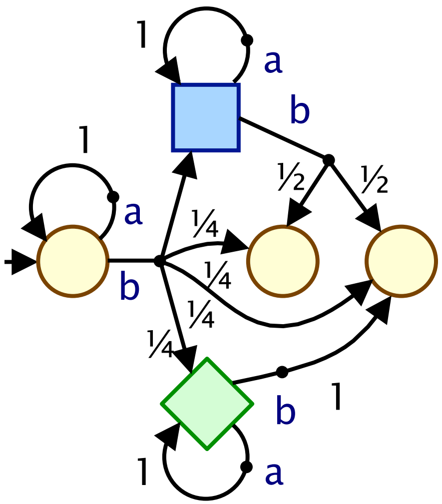
Assume-guarantee query:
"does component M_2 satisfy guarantee G, provided that assumption A always holds?"
reduces to…
"is there an adversary (strategy) of M_2 satisfying A but not G?"

Overview

- Probabilistic model checking & PRISM
  - example: Bluetooth
- Verification vs. strategy synthesis
  - Markov decision processes (MDPs)
  - example: robot controller
- Multi-objective probabilistic model checking
  - examples: team-formation/power management/…
- Model checking stochastic games
  - example: energy management

Stochastic multi-player games (SMGs)

- Stochastic multi-player games
  - players control states; choose actions
  - models competitive/collaborative behaviour
  - applications: security (system vs. attacker), controller synthesis (controller vs. environment), distributed algorithms/protocols, …

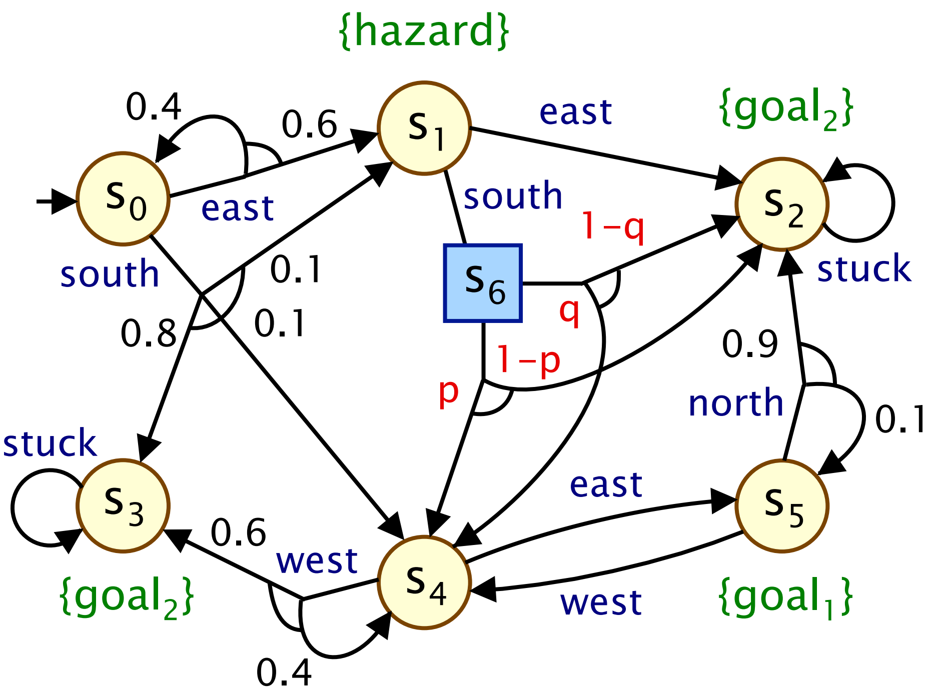


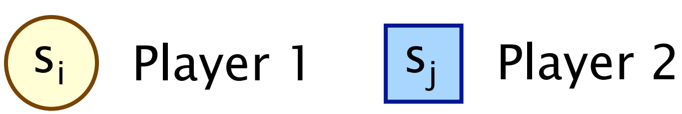
- Property specifications: rPATL
  - $\langle\langle\{1,2\}\rangle\rangle \, P_{\geq 0.95} [\, F^{\leq 45} \, done \,]$: "can nodes 1,2 collaborate so that the probability of the protocol terminating within 45 seconds is at least 0.95, whatever nodes 3,4 do?"
  - formally: $\langle\langle C\rangle\rangle\psi$: do there exist strategies for players in C such that, for all strategies of other players, property ψ holds?

- Model checking [TACAS'12,FMSD'13]
  - zero sum properties: analysis reduces to 2-player games
  - PRISM-games: www.prismmodelchecker.org/games

Example - Stochastic games

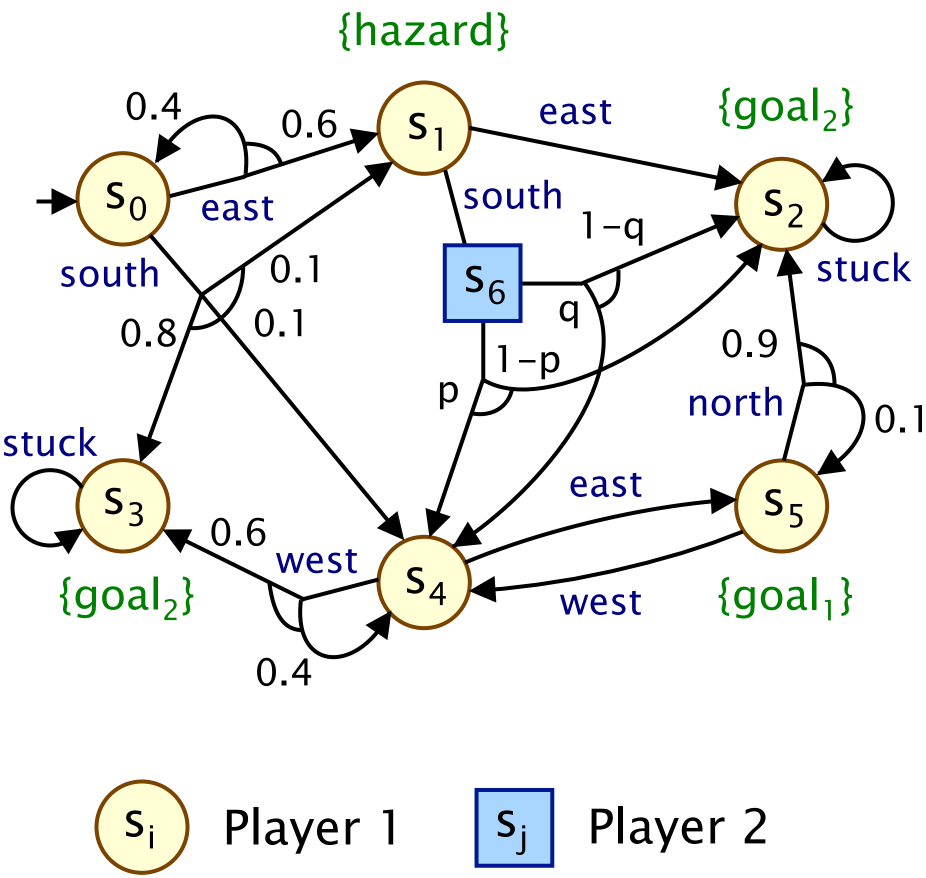
- Two players: 1 (robot controller), 2 (environment)
  - probability of s_1–south→s_4 is in [p,q] = [0.5–Δ, 0.5+Δ]





Example - Stochastic games

- Two players: 1 (robot controller), 2 (environment)
  - probability of s_1-south→s_4 is in [p,q] = [0.5-Δ, 0.5+Δ]



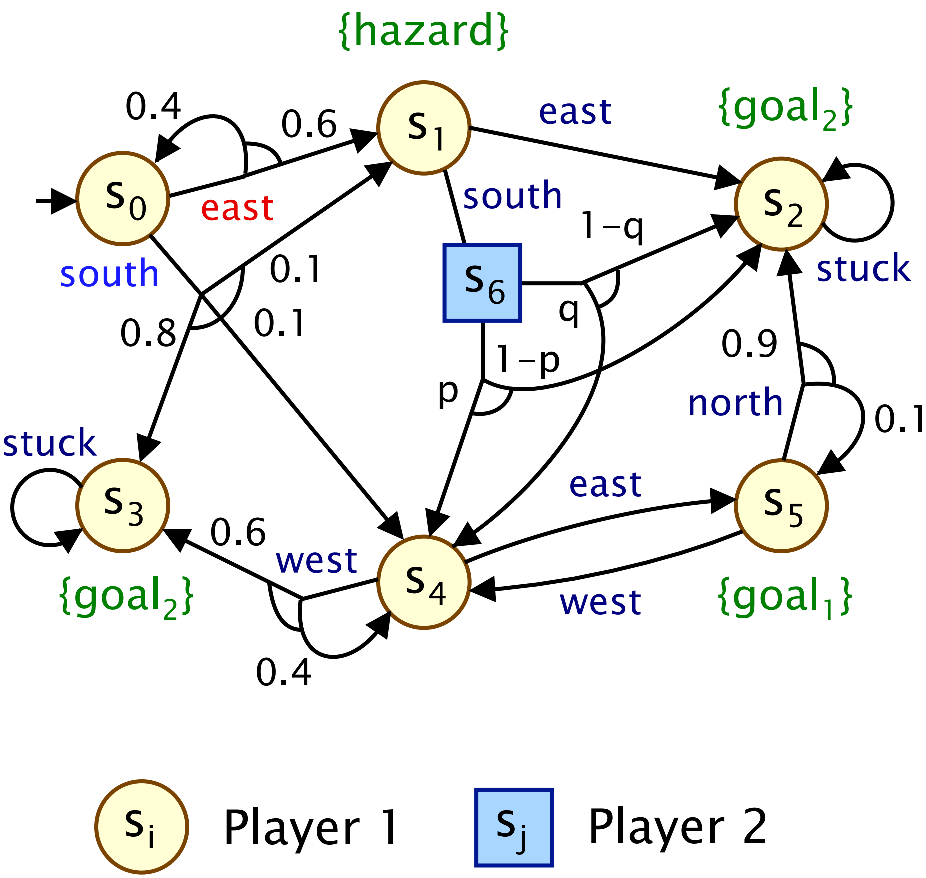
rPATL: $\langle\langle\{1\}\rangle\rangle\ P_{max=?}\ [\ F\ goal_1\ ]$

Optimal strategies:
memoryless and deterministic

Computation: graph analysis & numerical approximation

Example - Stochastic games

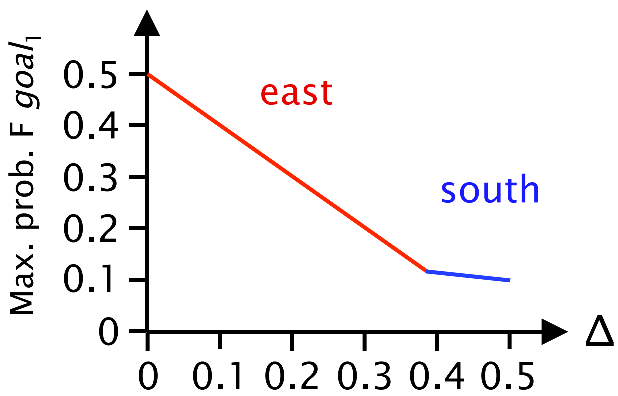
- Two players: 1 (robot controller), 2 (environment)
  - probability of s_1-south$\rightarrow s_4$ is in $[p,q] = [0.5-\Delta, 0.5+\Delta]$



rPATL: $\langle\langle\{1\}\rangle\rangle\, P_{max=?}[\, F\, goal_1\,]$

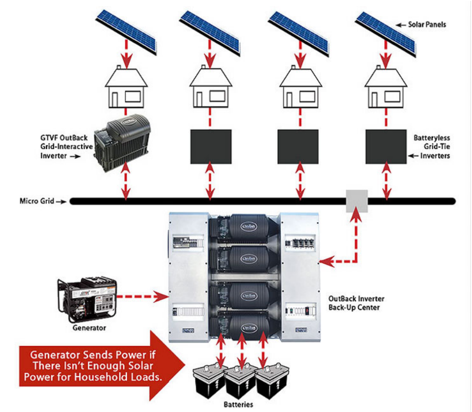
Optimal strategies:
memoryless and deterministic

Computation: graph analysis
& numerical approximation

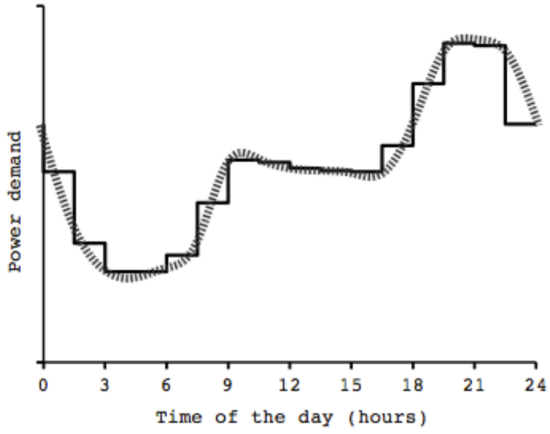


Example: Energy management

- Energy management protocol for Microgrid
  - Microgrid: local energy management
  - randomised demand management protocol
  - random back-off when demand is high



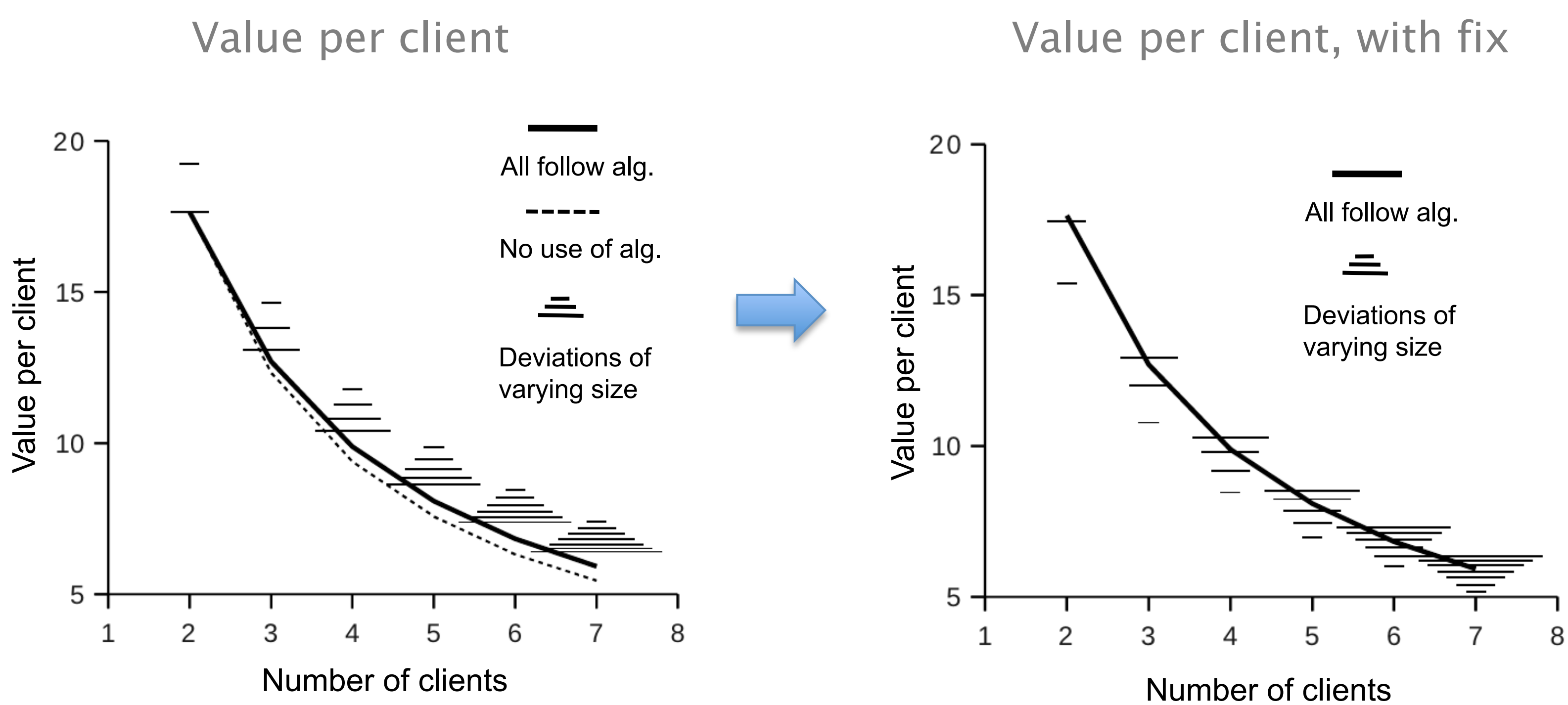
- Original analysis [Hildmann/Saffre'11]
  - protocol increases "value" for clients
  - simulation-based, clients are honest



- Our analysis
  - stochastic multi-player game model
  - clients can cheat (and cooperate)
  - model checking: PRISM-games

Example: Energy management

- Exposes protocol weakness
  - incentive for clients to act selfishly
- We propose a simple fix (and verify it)
  - clients can be punished



Conclusion

- Probabilistic model checking & PRISM
  - Markov decision processes (MDPs)
  - PCTL, probabilistic LTL, expected costs/rewards
  - verification vs. controller synthesis
- Multi-objective probabilistic model checking
  - trade-offs between conflicting objectives
  - achievability queries, numerical queries, Pareto curves
- Model checking for stochastic multi-player games
  - competitive/collaborative behaviour
  - rPATL model checking

– Challenges
  - stochastic games: multiple objectives, richer temporal logics
  - partial information/observability