

# Mean-Payoff Optimization in Continuous-Time Markov Chains with Parametric Alarms

Christel Baier, Clemens Dubslaff,  
Ľuboš Korenčiak, Antonín Kučera, Vojtěch Řehák

TU Dresden, Germany  
Faculty of Informatics, Masaryk University, Brno, Czech Republic

14th September 2017

Published in Quantitative evaluation of systems (QEST) 2017

# The Aim

Parametric model,  $\mathbf{p} \in D$

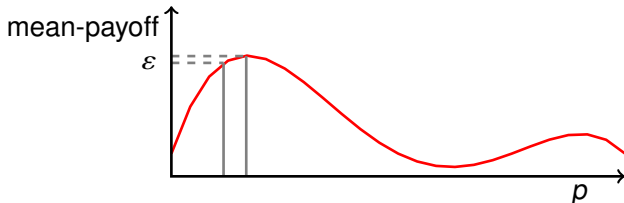
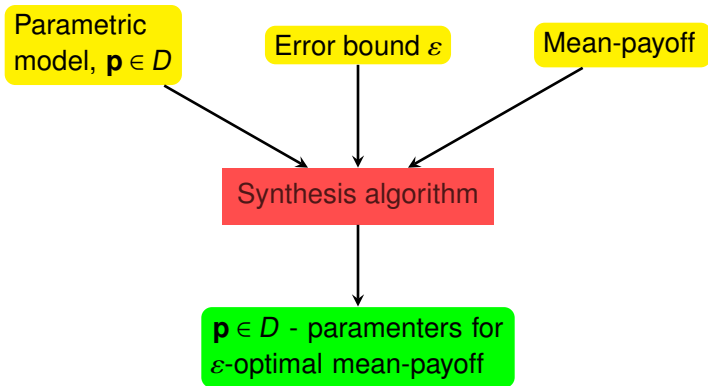
Mean-payoff

Synthesis algorithm

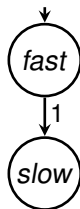
$\mathbf{p} \in D$  - parameters for optimal mean-payoff



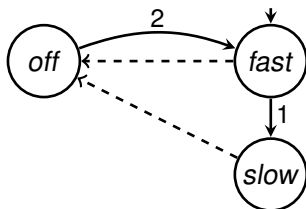
# The Aim



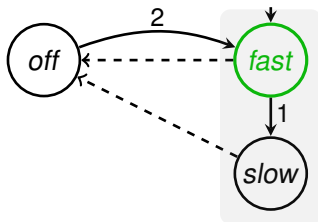
# Continuous-time Markov Chain with Alarms (ACTMC)



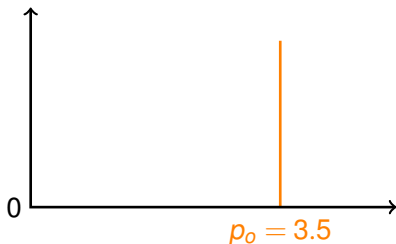
# Continuous-time Markov Chain with Alarms (ACTMC)



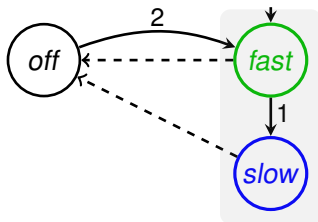
# Continuous-time Markov Chain with Alarms (ACTMC)



$$\omega = (o=3.5, s=2),$$



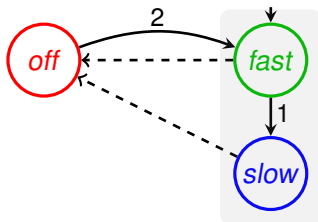
# Continuous-time Markov Chain with Alarms (ACTMC)



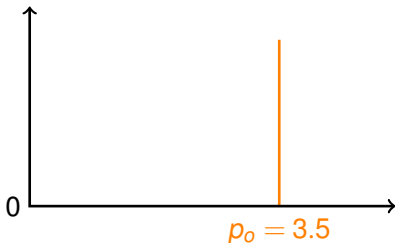
$$\omega = (o=3.5, s=2), (o=1.5),$$



# Continuous-time Markov Chain with Alarms (ACTMC)

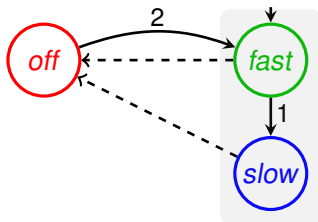


$$\omega = (o=3.5, s=2), (o=1.5), (f=4), (\dots), \dots$$

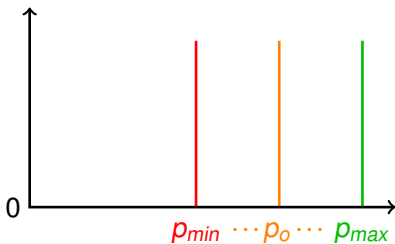




# Parametric ACTMC

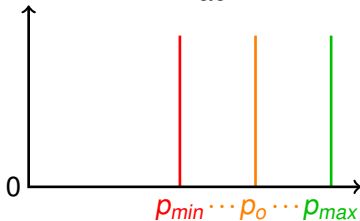


$$\omega = (o=3.5, s=2), (o=1.5), (f=4), (\dots), \dots$$

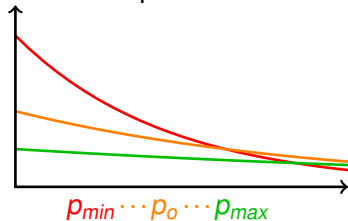


# Parametric distributions

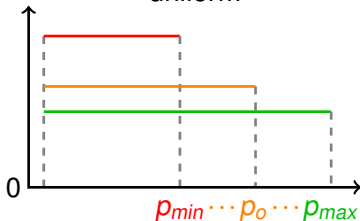
Dirac



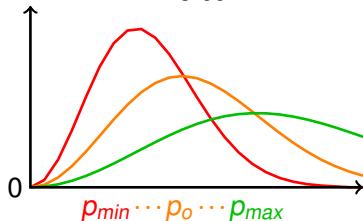
exponential



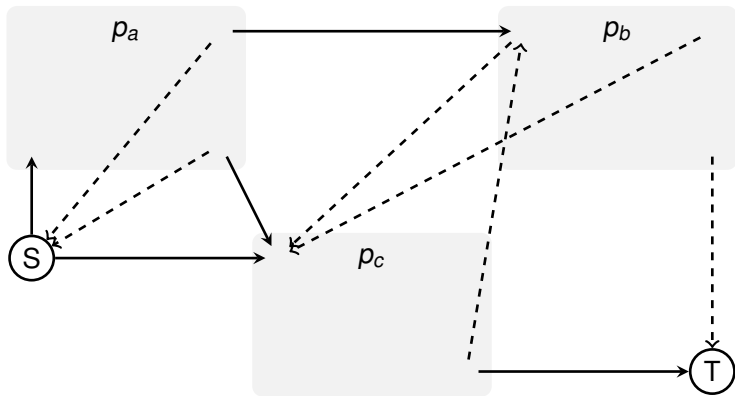
uniform



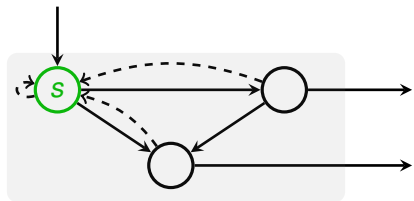
Weibull



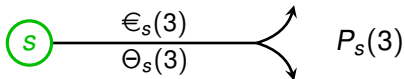
# Restrictions - disjoint regions



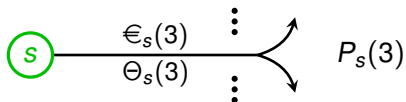
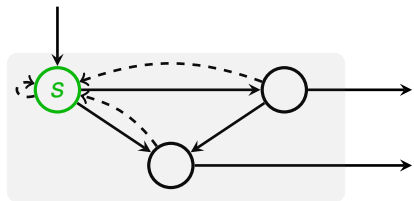
# Explicit Algorithm



ACTMC and  $\varepsilon$ ;  $\mathbf{p} = ?$



# Explicit Algorithm

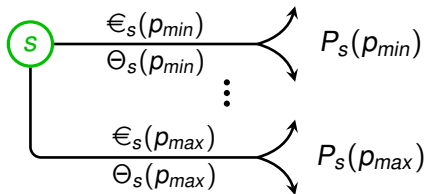
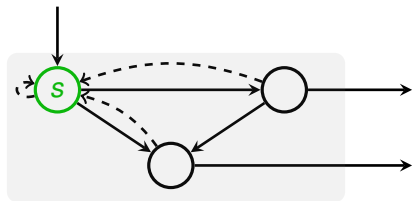


ACTMC and  $\epsilon$ ;  $\mathbf{p} = ?$



uncountable action  
space MDP

# Explicit Algorithm



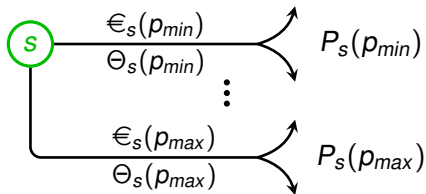
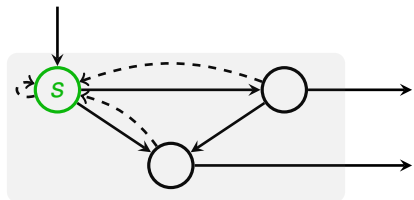
ACTMC and  $\epsilon$ ;  $\mathbf{p} = ?$

↓  
uncountable action  
space MDP

↓  
**finite MDP**

↓  
 $\epsilon$ -optimal  $\mathbf{p}$

# Explicit Algorithm



ACTMC and  $\epsilon$ ;  $\mathbf{p} = ?$

uncountable action  
space MDP

finite MDP

$\epsilon$ -optimal  $\mathbf{p}$

For  $\epsilon = 0.1$  we get at least  $10^{20}$  actions.

Our contribution is based on two ideas:

- policy iteration
- symbolic representation of actions

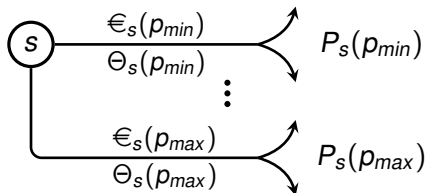


# Symbolic Algorithm

## policy iteration

$$\arg \max_{p \in D} F_s(p)$$

$$F_s(p) = \dots \epsilon_s(p) \dots \Theta_s(p) \dots P_s(p) \dots$$



ACTMC and  $\epsilon$ ;  $\mathbf{p} = ?$

↓  
uncountable action  
space MDP

↓  
finite MDP

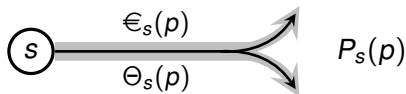
↓  
 $\epsilon$ -optimal  $\mathbf{p}$

# Symbolic Algorithm

## policy iteration

$$\arg \max_{p \in D} F_s(p)$$

$$F_s(p) = \dots \epsilon_s(p) \dots \Theta_s(p) \dots P_s(p) \dots$$



### Message:

Sometimes it is better to use **continuous** approach than finite discretization.

ACTMC and  $\epsilon$ ;  $\mathbf{p} = ?$

↓  
uncountable action  
space MDP

↓  
MDP with  
symbolic actions

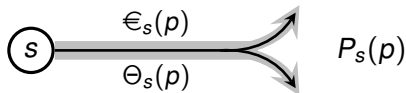
↓  
 $\epsilon$ -optimal  $\mathbf{p}$

# Symbolic Algorithm

## policy iteration

$$\arg \max_{p \in D} F_s(p)$$

$$F_s(p) = \dots \epsilon_s(p) \dots \Theta_s(p) \dots P_s(p) \dots$$



### Message:

Sometimes it is better to use **continuous** approach than finite discretization.

ACTMC and  $\epsilon$ ;  $\mathbf{p} = ?$

↓  
uncountable action  
space MDP

↓  
MDP with  
symbolic actions

↓  
 $\epsilon$ -optimal  $\mathbf{p}$

Thank you for your attention!