Bayes-Adaptive Planning for Data-Efficient Verification of Uncertain Markov Decision Processes

Viraj Brian Wijesuriya & Alessandro Abate

Department of Computer Science, University of Oxford

16th International Conference on Quantitative Evaluation of SysTems (QEST 2019), Glasgow, UK
10th September 2019

Outline

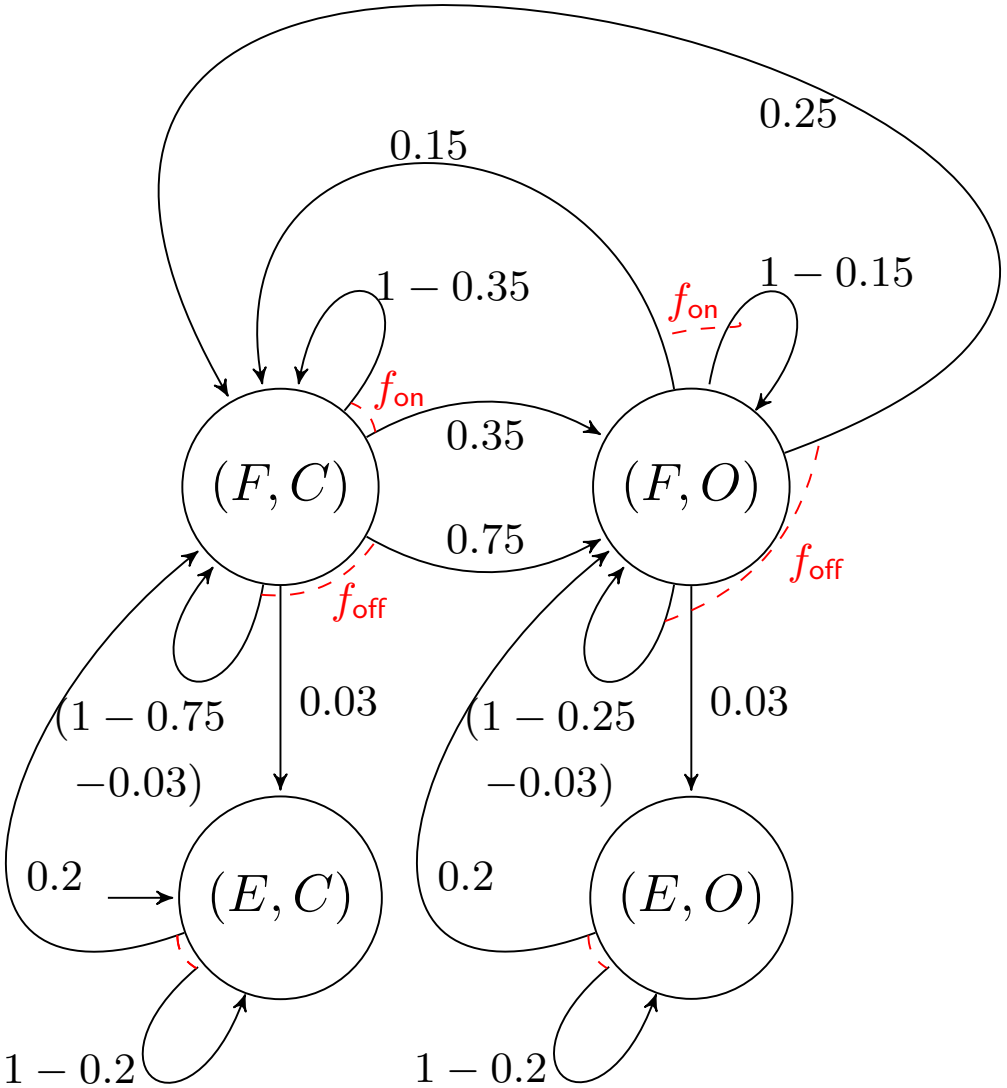
Automated Formal Verification - New Frontiers

- Verification of physical systems
  - Need to reduce the gap between data and models
  - Principled integration of learning and verification

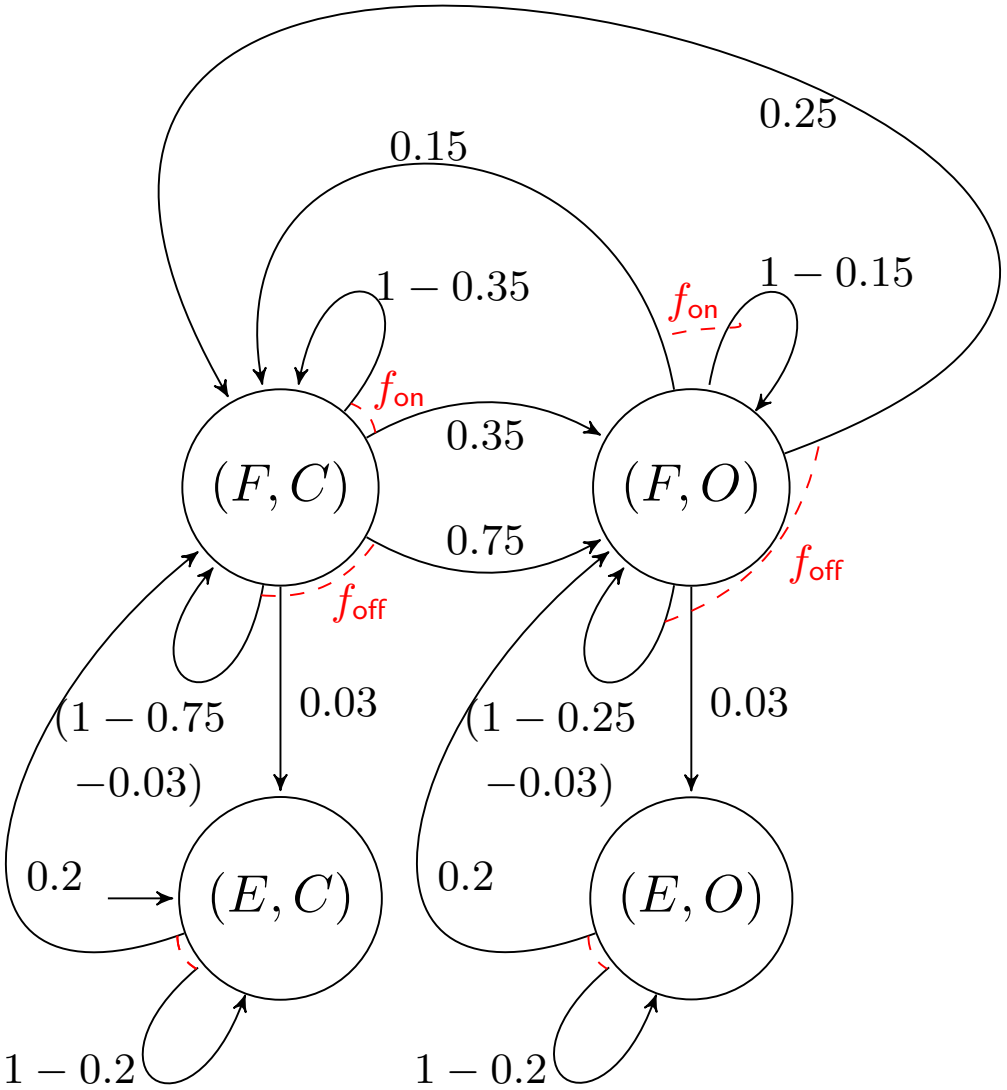
Figure: Advanced modelling for smart buildings: Building automation system setup in rooms 478/9 at Oxford CS with Estate Services at Oxford, Honeywell (CZ), Trend Control (UK)

Applications: control of temperature, humidity, CO_2, model-based predictive maintenance of devices, fault-tolerant certified control, demand-response over smart grids

Discrete-time Markov Decision Process (MDP)



Discrete-time Markov Decision Process (MDP)



transition probability function

finite set of actions

finite set of initial states

finite set of states

$$\mathcal{M} = (S, A, \mathcal{T}, \iota)$$

Windows are open (O) or closed (C)
Zone is occupied (F) or not (E)

$$\mathcal{T} \colon S \times A \times S \to [0, 1]$$

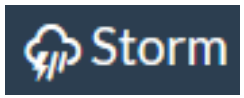
$$\mathcal{T} = \left[\begin{array}{cccc} 0.8 & 0.2 & 0 & 0 \\ \hline 0 & 0.65 & 0.35 & 0 \\ 0.03 & 0.22 & 0.75 & 0 \\ \hline 0 & 0.15 & 0.85 & 0 \\ 0 & 0.25 & 0.72 & 0.03 \\ \hline 0 & 0 & 0.2 & 0.8 \end{array}\right]$$

Probabilistic Computational Tree Logic (PCTL) Properties

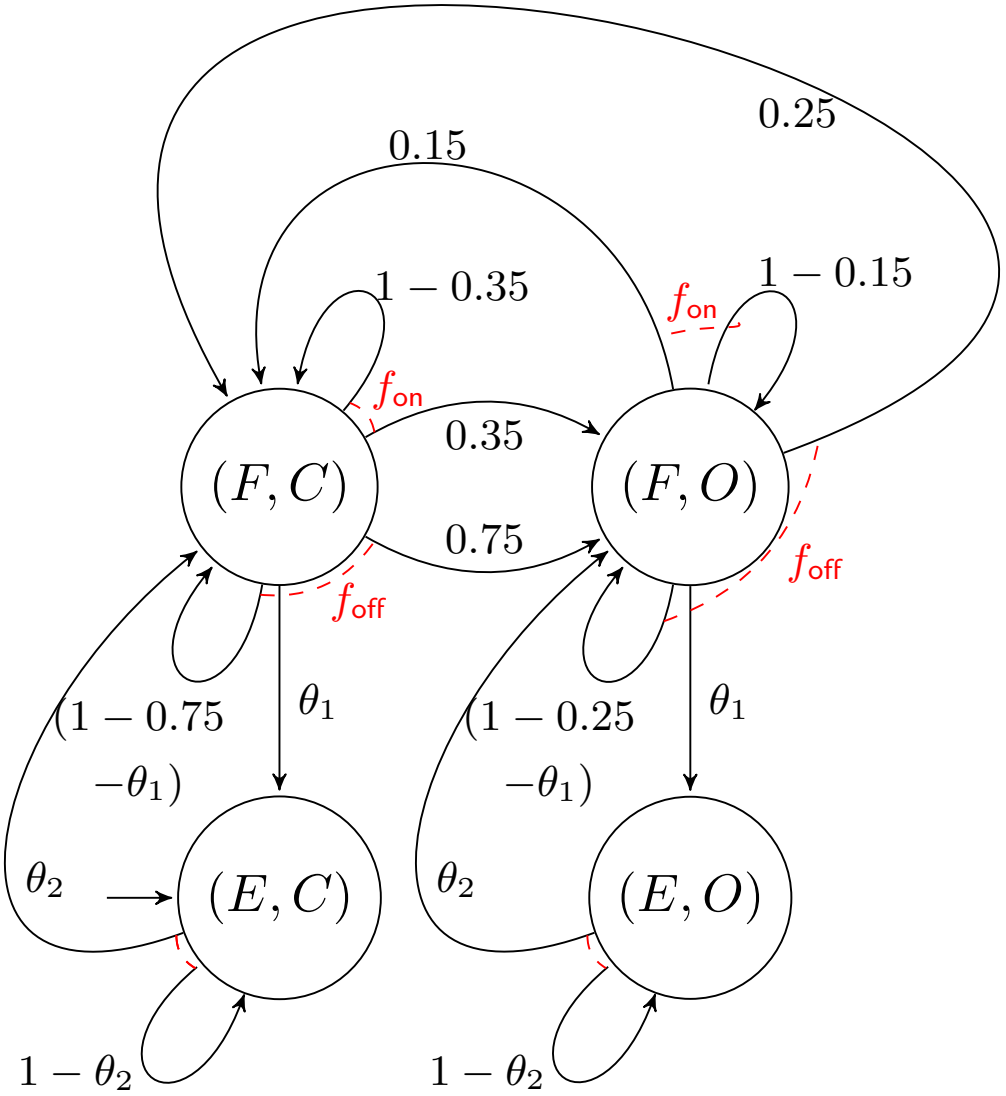
$non - nested$ PCTL properties compatible with Storm

$\mathbf{P}_{\geq 0.35} \neg (\text{true}\ \mathcal{U}^{\leq 20}\ (\text{E}, \text{O}))$

$\mathbf{P}_{\geq 0.5} (\text{true}\ \mathcal{U}\ (\text{E}, \text{O}))$



Uncertain Markov Decision Process



Parametric Markov Decision Process (pMDP)

finite set of states — finite set of actions — finite set of initial states

$$\mathcal{M}_p = (S, A, \mathcal{T}_p, \iota, \Theta)$$

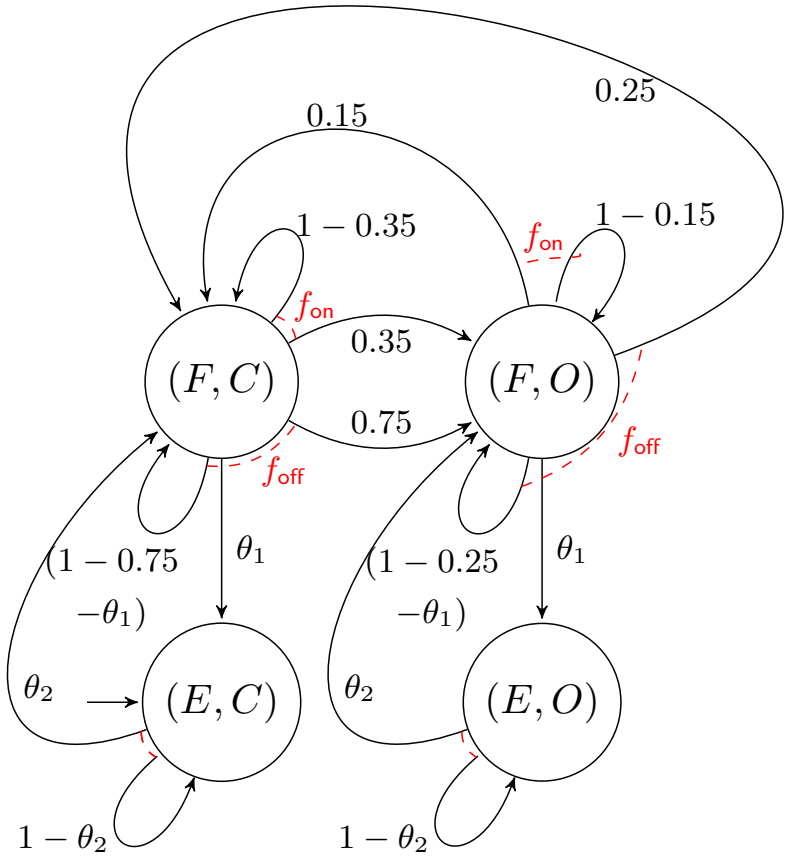
$$\mathcal{T}_p = \left[\begin{array}{cccc} 1-\theta_2 & \theta_2 & 0 & 0 \\ \hline 0 & 0.65 & 0.35 & 0 \\ \theta_1 & 1-0.75-\theta_1 & 0.75 & 0 \\ \hline 0 & 0.15 & 0.85 & 0 \\ 0 & 0.25 & 1-0.25-\theta_1 & \theta_1 \\ \hline 0 & 0 & \theta_2 & 1-\theta_2 \end{array}\right]$$

$$\Theta \subseteq [0,1]^n = [0, 0.25] \times [0, 1]$$

Parameter synthesis: find values for parameters $\theta \in \Theta$ s.t. induced model $\mathcal{M}(\theta)$ satisfies φ:

$$\Theta_\varphi = \{\theta \in \Theta : M(\theta) \models \varphi\} \subseteq \Theta$$

We call Θ_φ the *feasible set*.

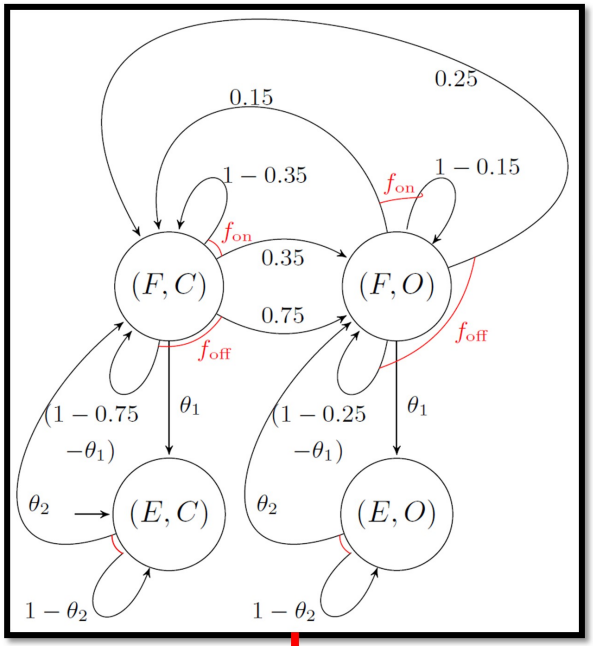


Parameter Synthesis

PCTL

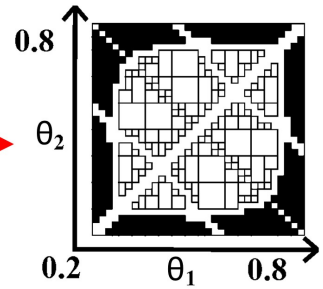
$\varphi = P_{>=0.5}[\text{true } \mathcal{U} \text{ (E,O)}]$

System



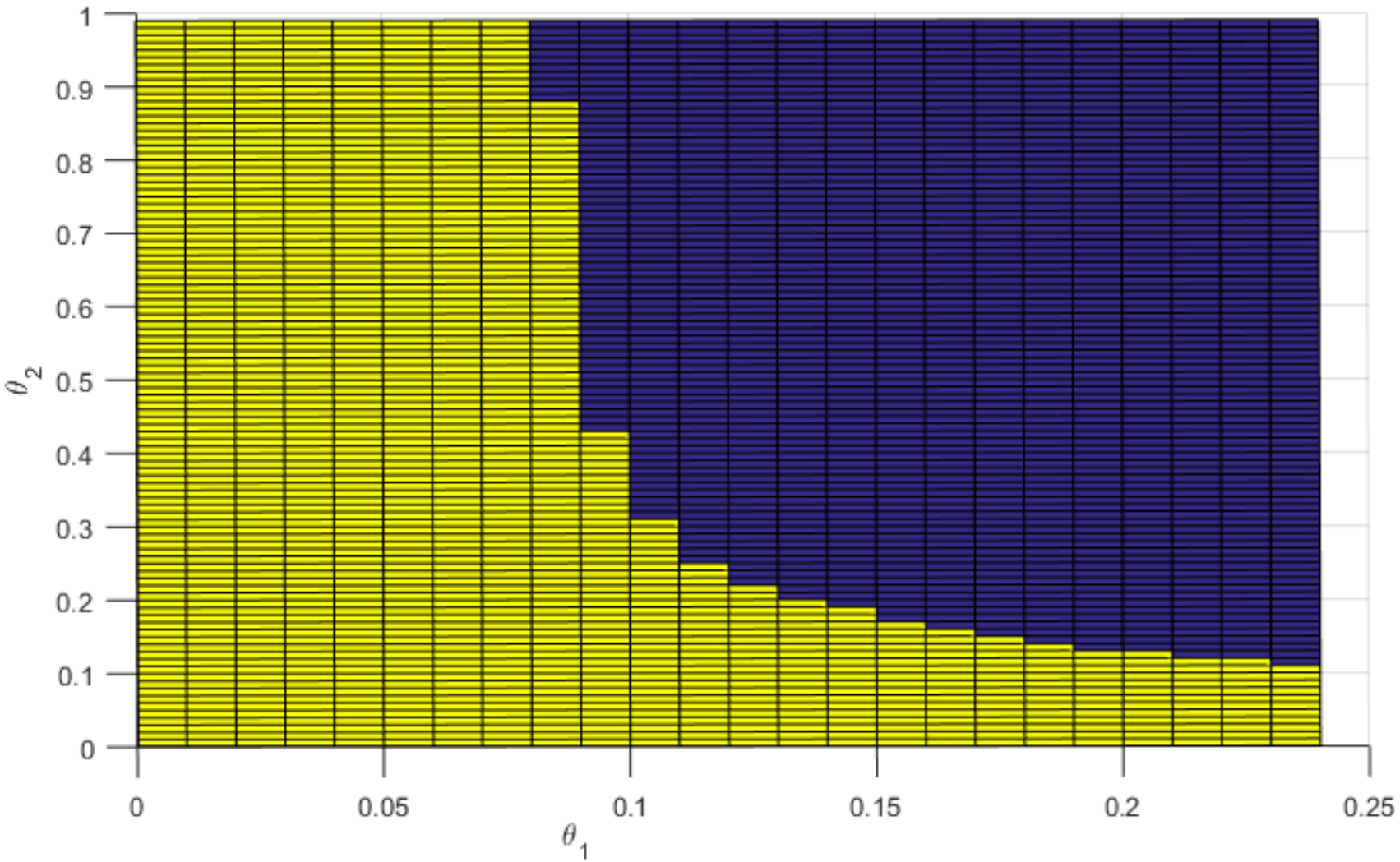
pMDP

Probabilistic Model Checker



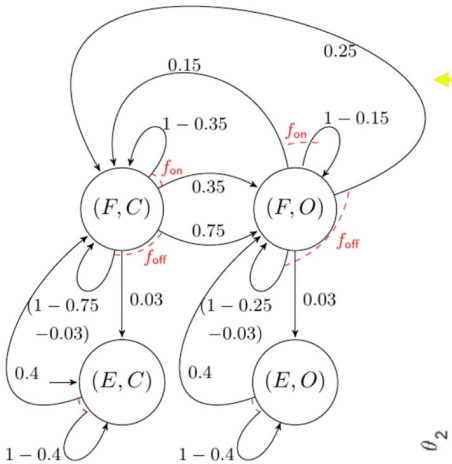
Regions ➔ Truth Values

Parameter Synthesis



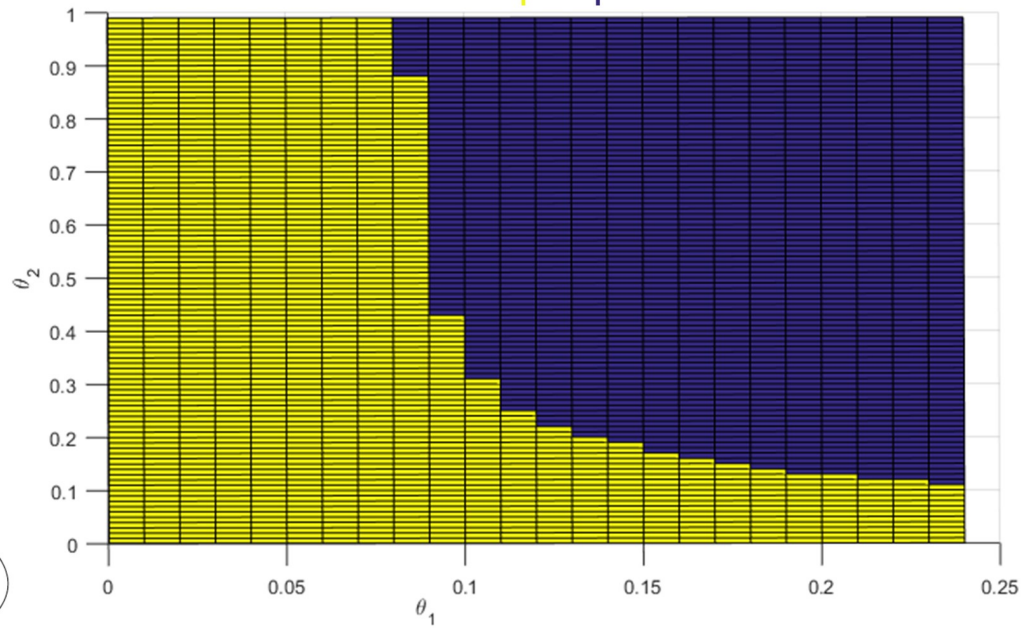
For the property $\varphi = \mathbf{P}_{>0.3}(\Box^{\leq 20}\ \neg(\mathsf{E}, \mathsf{O}))$, synthesis yields the feasible set Θ_φ (yellow set) which contains all valuations of parameters that satisfy φ

Model Classification



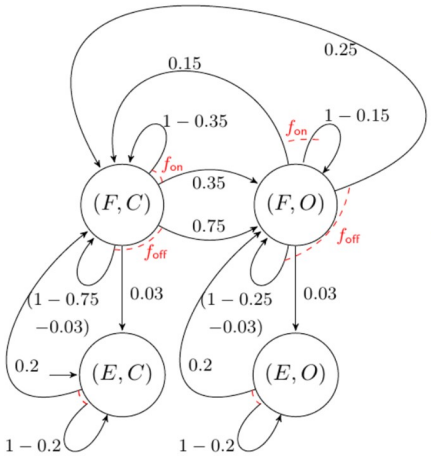
$S \models \varphi$

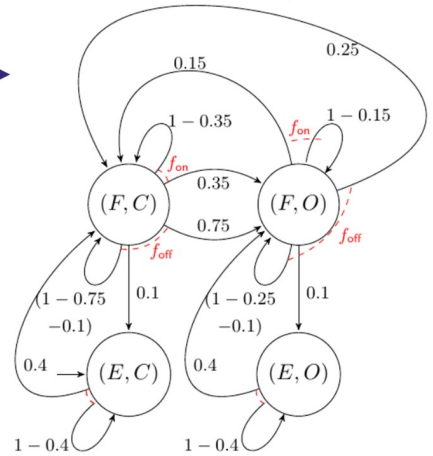
$S \not\models \varphi$

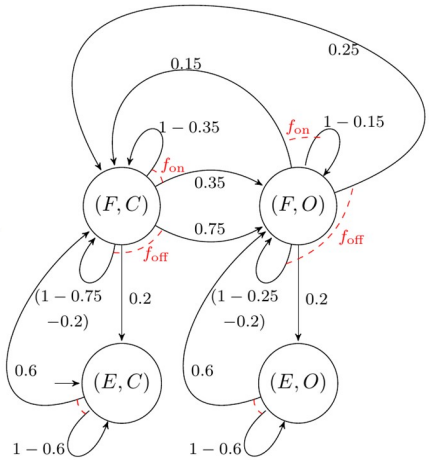


$S \models \varphi$

$S \not\models \varphi$







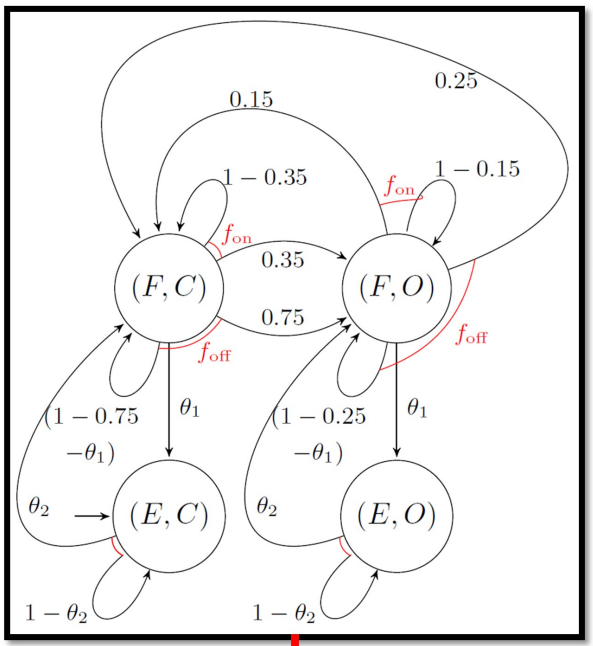
System Verification

Does the system satisfy the property?

PCTL

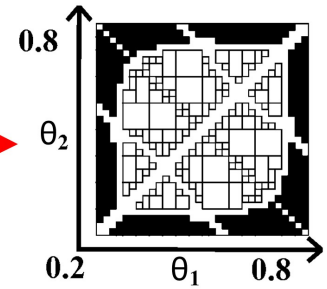
$\varphi = P_{>=0.5}[\text{true } \mathcal{U} \text{ (E,O)}]$

System



pMDP

Probabilistic Model Checker

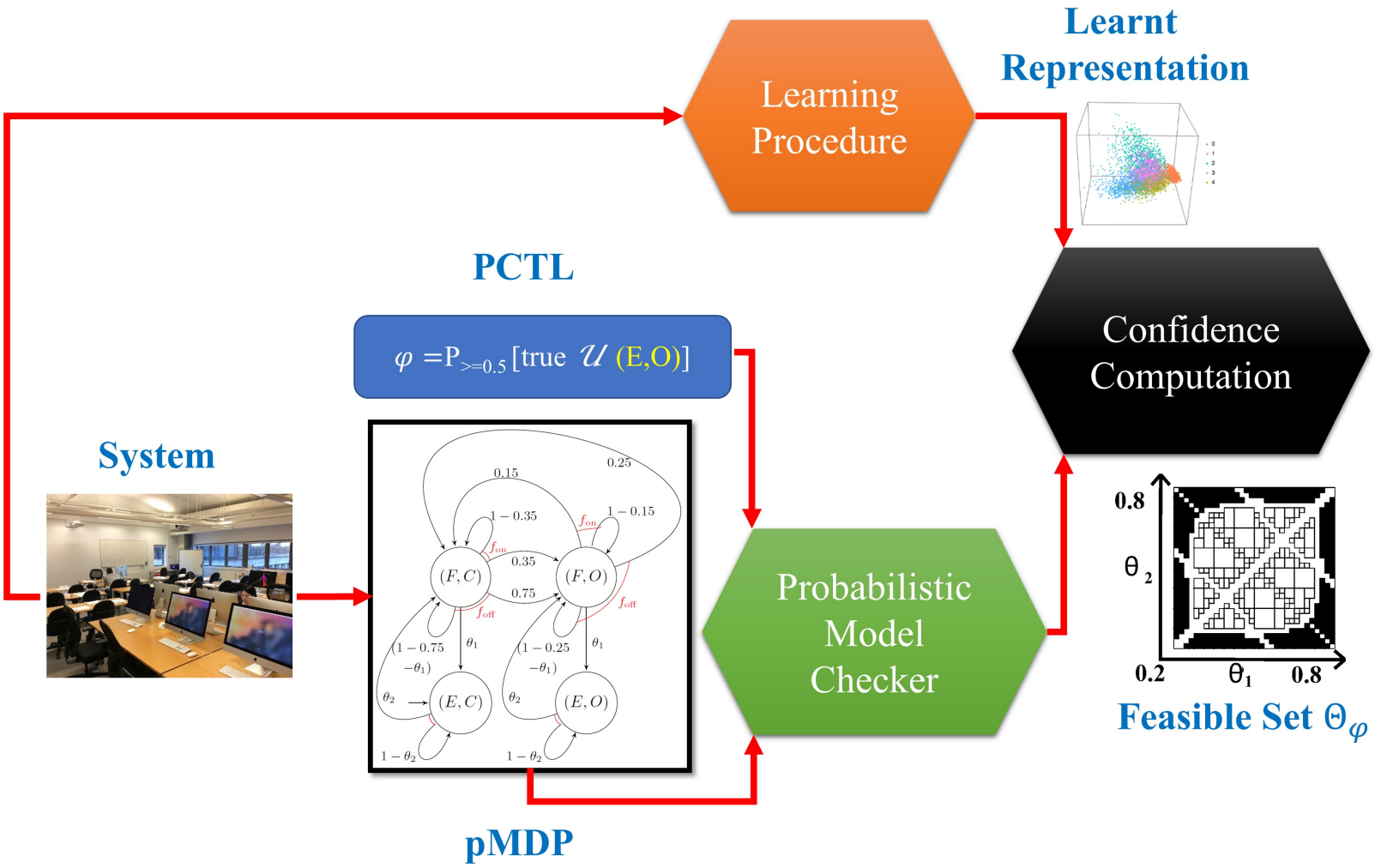


Regions ➔ Truth Values

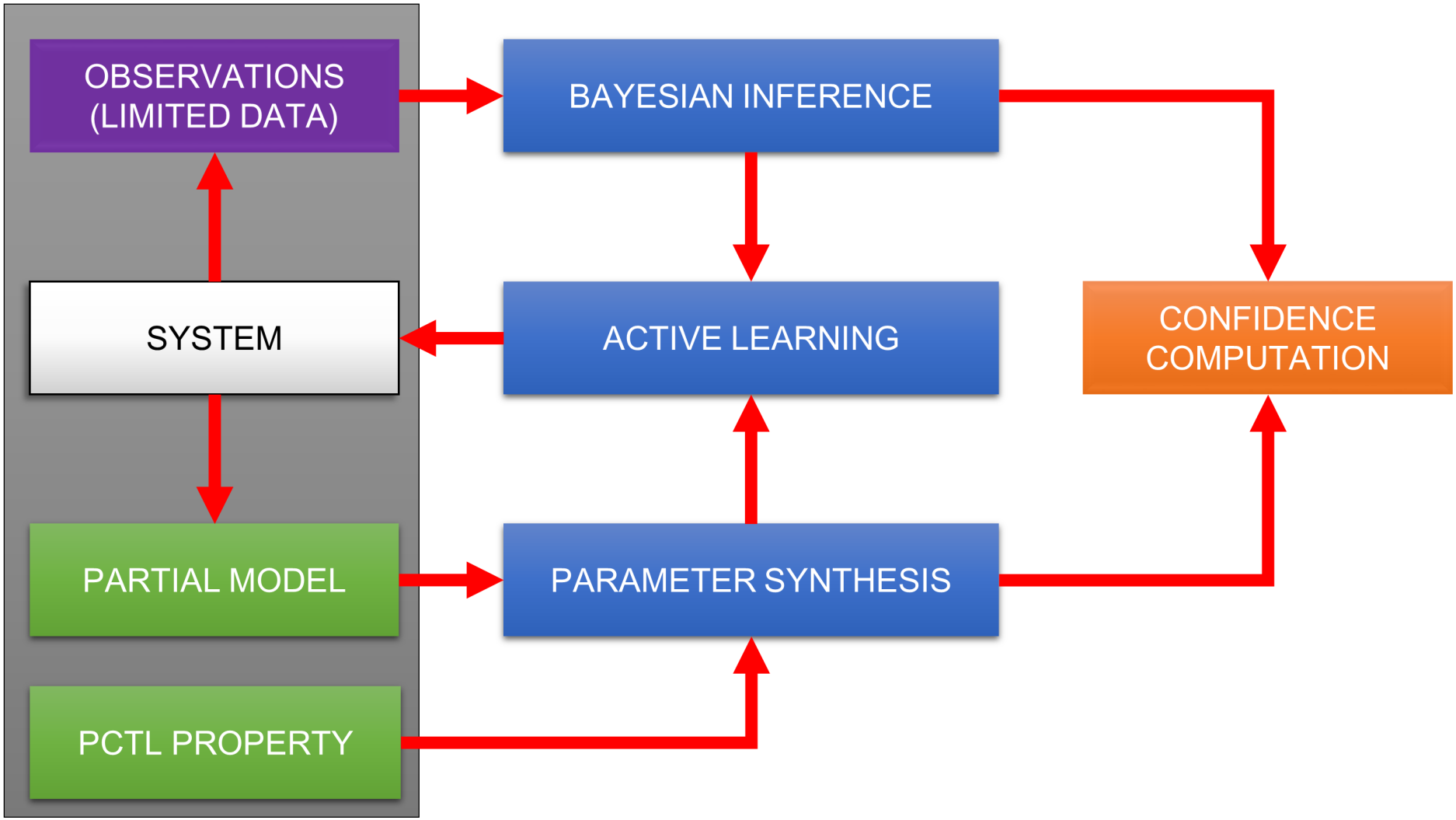
Outline

Integrated Learning and Verification Framework

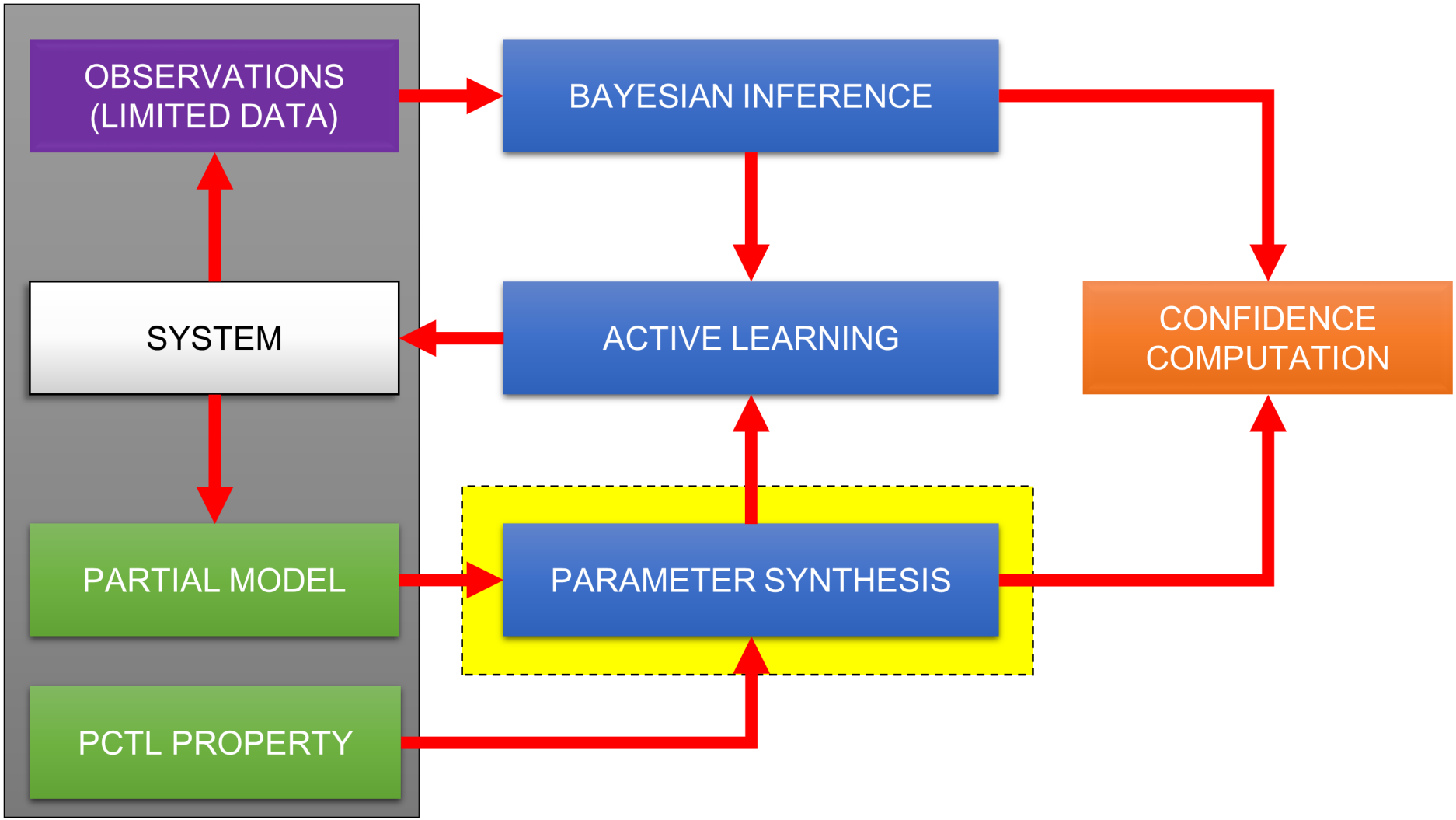
Incomplete knowledge in pMDP can complement the learning process



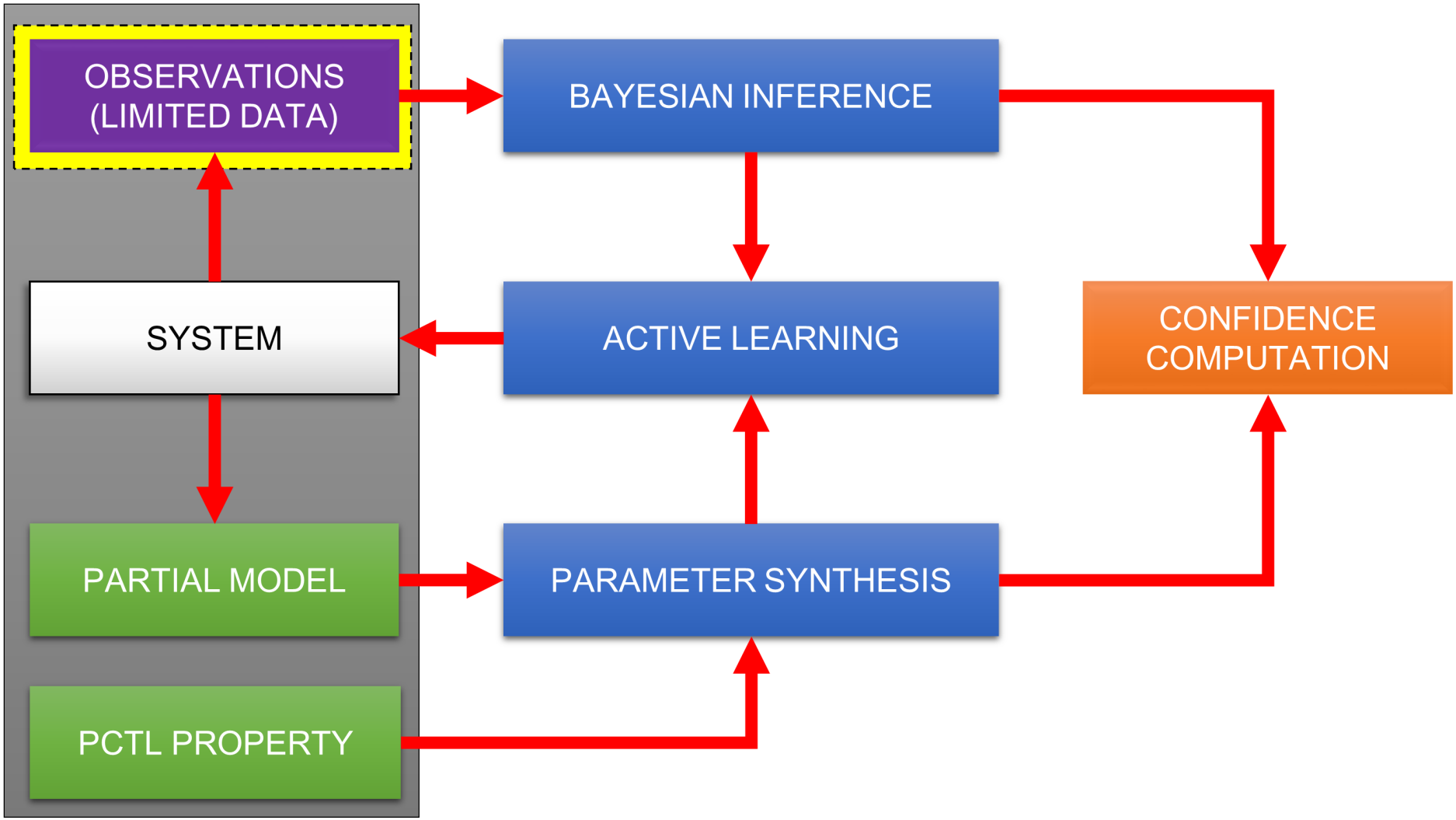
Overview


OBSERVATIONS
(LIMITED DATA)
BAYESIAN INFERENCE
SYSTEM
ACTIVE LEARNING
CONFIDENCE
COMPUTATION
PARTIAL MODEL
PARAMETER SYNTHESIS
PCTL PROPERTY

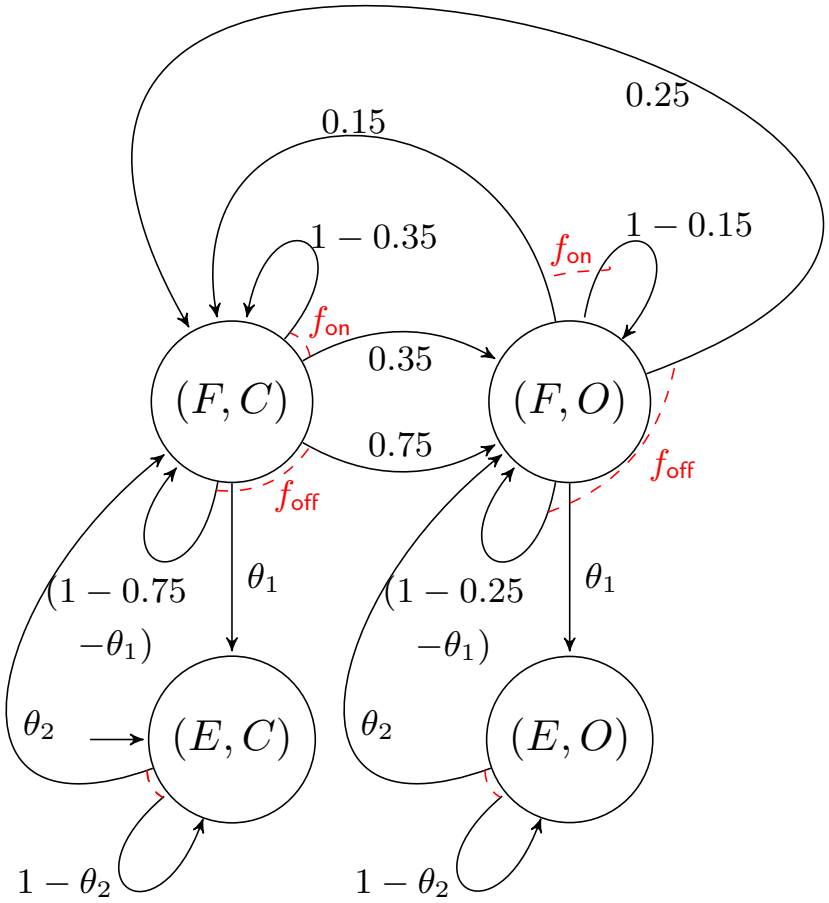
Overview


OBSERVATIONS
(LIMITED DATA)
BAYESIAN INFERENCE
SYSTEM
ACTIVE LEARNING
CONFIDENCE
COMPUTATION
PARTIAL MODEL
PARAMETER SYNTHESIS
PCTL PROPERTY

Overview


OBSERVATIONS
(LIMITED DATA)
BAYESIAN INFERENCE
SYSTEM
ACTIVE LEARNING
CONFIDENCE
COMPUTATION
PARTIAL MODEL
PARAMETER SYNTHESIS
PCTL PROPERTY

Observations/Data



Group transition counts by parameter for $\exists \bar{s} \in S : \mathcal{T}_p(s,a,\bar{s}) = \theta_i$

$$C(\theta_i) = \sum C^a_{s,s'} \text{ for } \mathcal{T}_p(s,a,s') = \theta_i$$

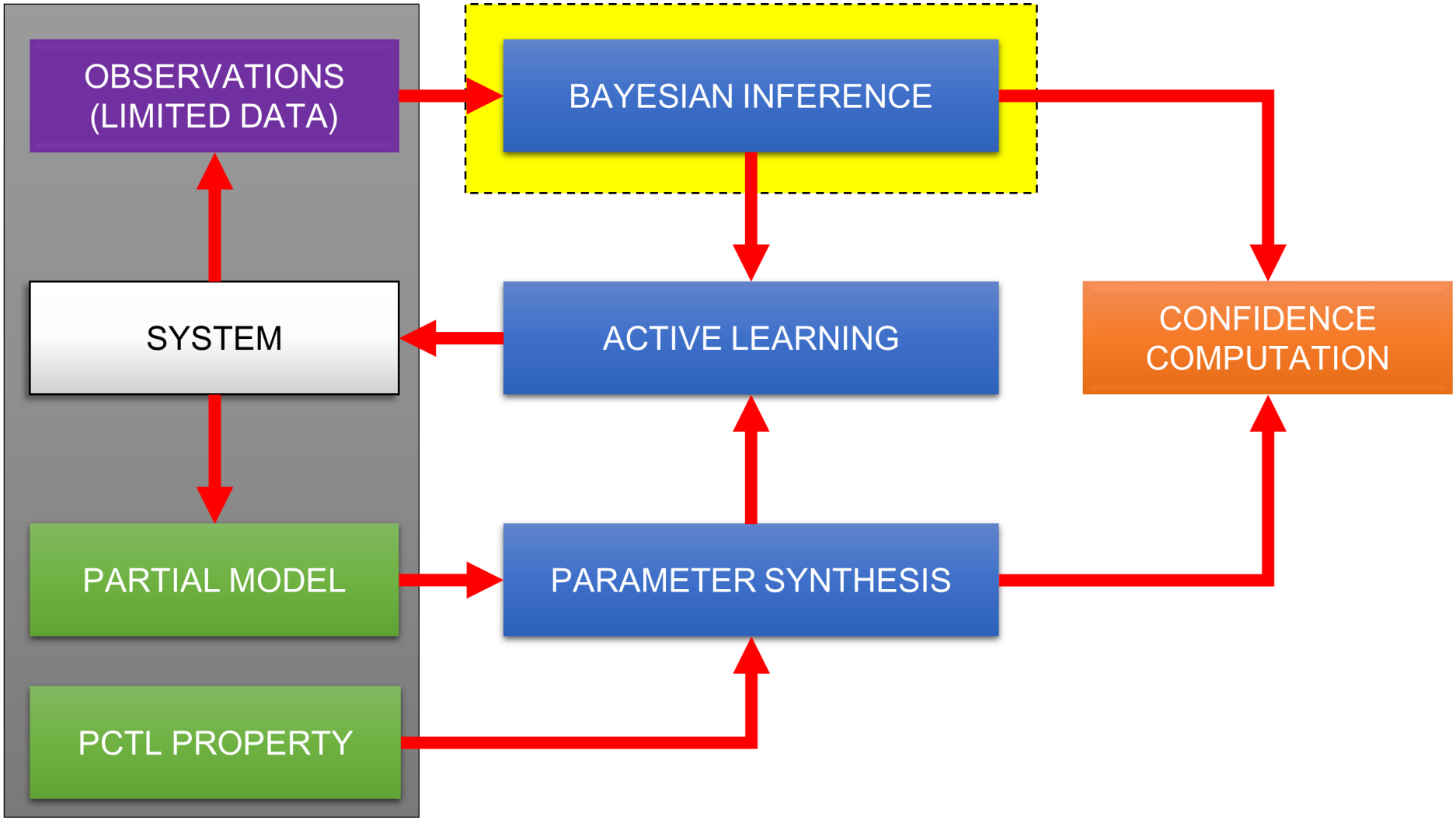
$$C'(\theta_i) = \sum C^a_{s,s'} \text{ for } \mathcal{T}_p(s,a,s') \neq \theta_i$$

$$\bar{C}(\theta_i) = C(\theta_i), C'(\theta_i)$$

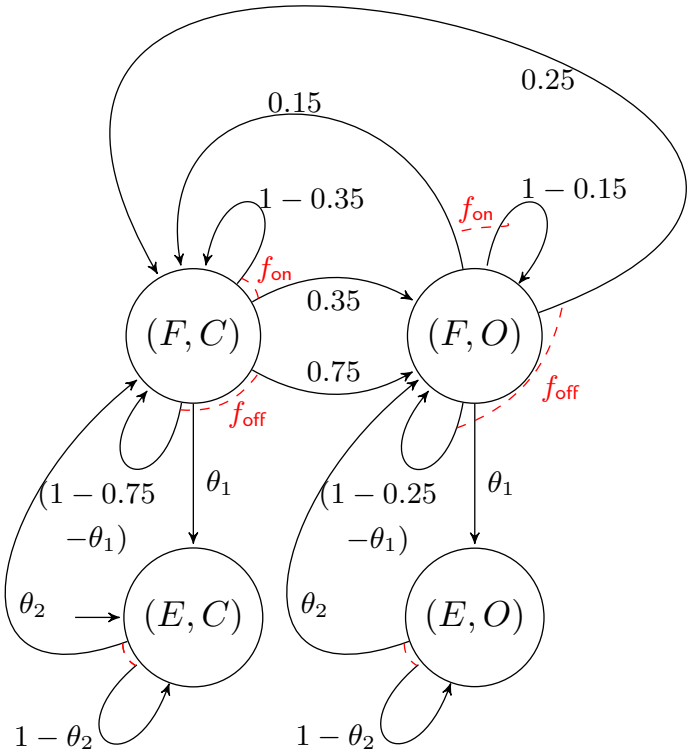
Trace/Trajectory :

$$(\{E,C\},\cdot) \xrightarrow{\theta_2} (\{F,C\},f_{\text{off}}) \xrightarrow{1-0.75-\theta_1} (\{F,C\},f_{\text{off}}) \xrightarrow{0.75} (\{F,O\},f_{\text{off}}) \xrightarrow{0.25} (\{F,C\},f_{\text{off}}) \xrightarrow{\theta_1} (\{E,C\},\cdot) \xrightarrow{1-\theta_2} (\{E,C\},\cdot)\ldots$$

Overview


OBSERVATIONS
(LIMITED DATA)
BAYESIAN INFERENCE
SYSTEM
ACTIVE LEARNING
CONFIDENCE
COMPUTATION
PARTIAL MODEL
PARAMETER SYNTHESIS
PCTL PROPERTY

Bayesian Inference



$$P(\theta_i \mid \mathscr{D}) = \frac{P(\mathscr{D} \mid \theta_i)P(\theta_i)}{P(\mathscr{D})}$$

observed — prior — binomial

$$= \frac{P(\theta_i)\theta_i^{C(\theta_i)}(1-\theta_i)^{C'(\theta_i)}}{P(\bar{C}(\theta_i))}$$

Select a conjugate prior - Beta distribution:

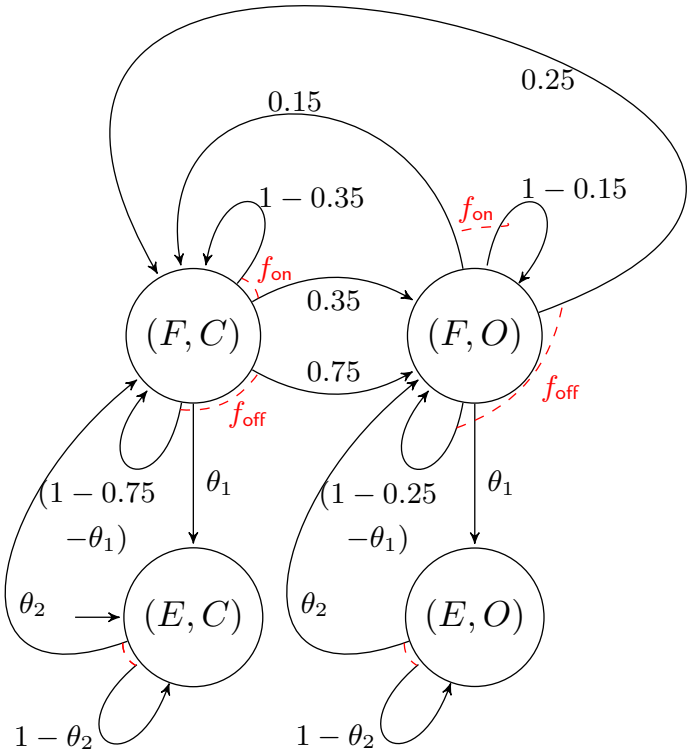
$$P(\theta_i) = Beta(\theta_i;\omega) \propto \theta_i^{\alpha-1}(1-\theta_i)^{\beta-1}$$

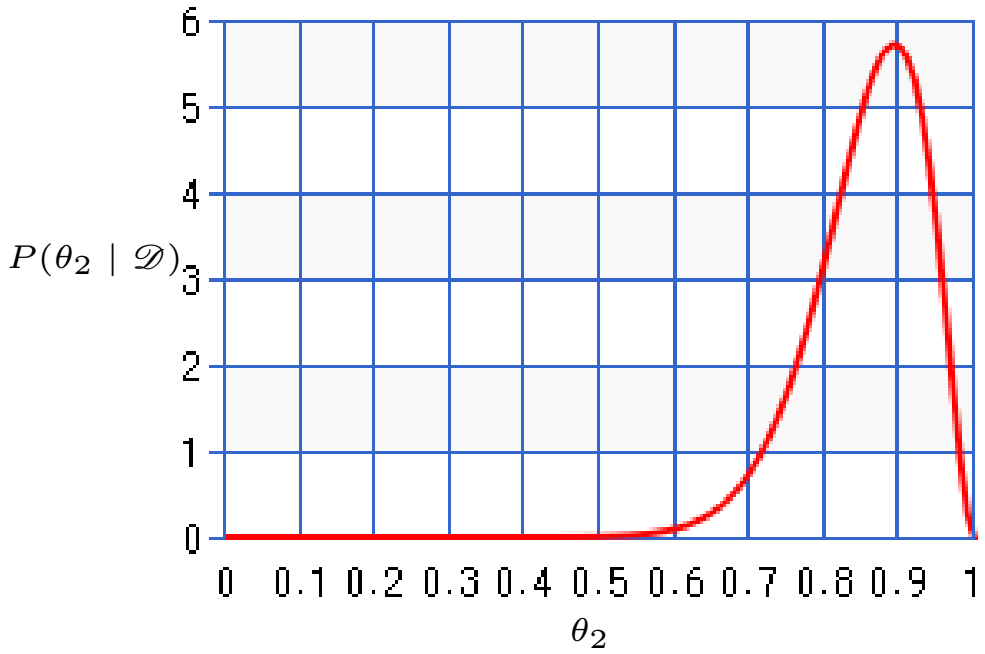
for pair $(\theta_i, 1-\theta_i)$, with $\omega = (\alpha, \beta)$ hyperparameters

Under Beta prior, posterior update is analytic:

$$P(\theta_i \mid \mathscr{D}) = Beta(\theta_i; \bar{C}(\theta_i) + \omega) \propto \theta_i^{C(\theta_i)}(1-\theta_i)^{C'(\theta_i)}$$

Bayesian Inference





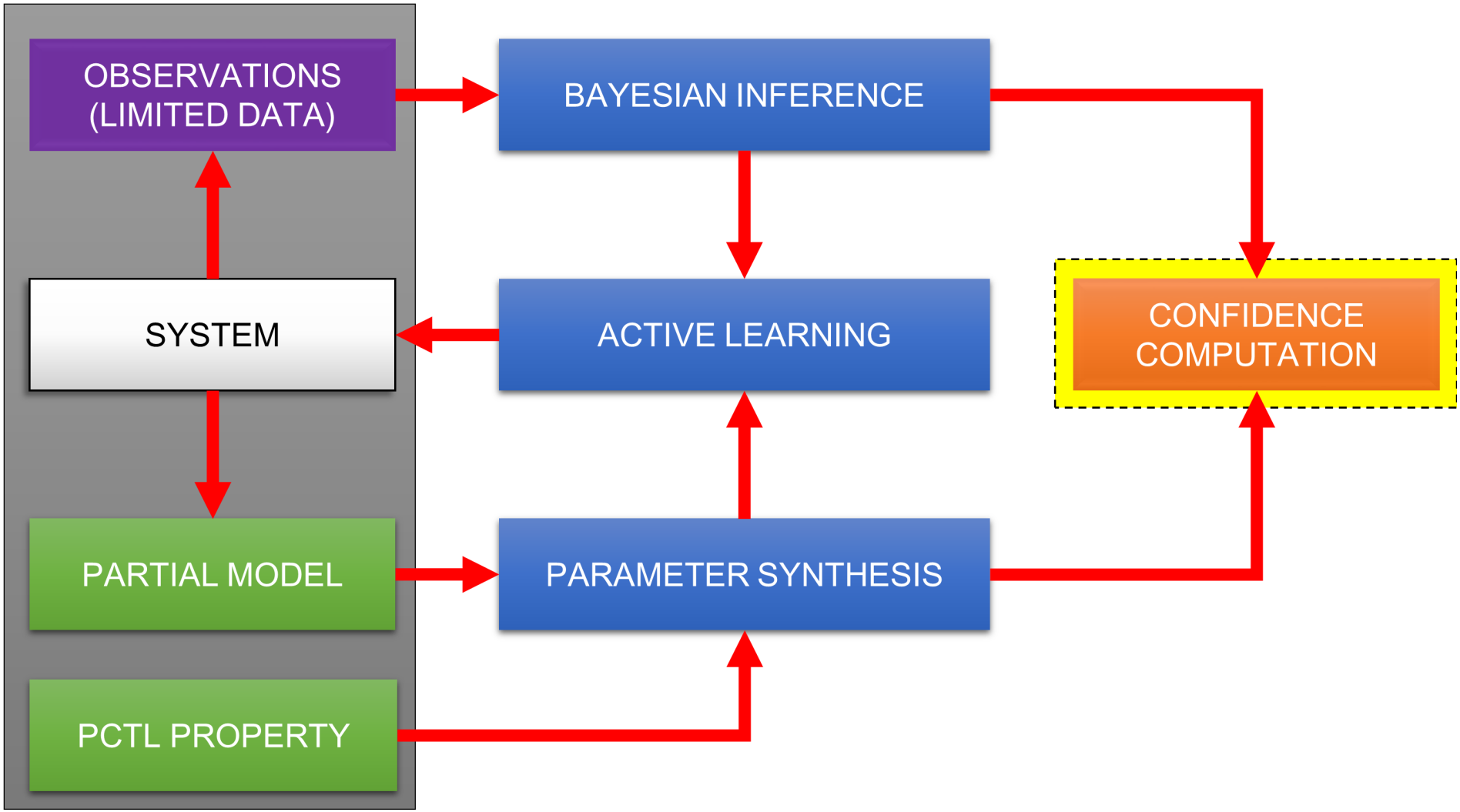
With prior $\alpha = (1, 1)$

$P(\theta_2 \mid \mathscr{D}) = Beta(\theta_2; 18, 3)$

Consider (E, C) and (E, O)

- Observe $\theta_2 \rightarrow C(\theta_2) = 17$
- Observe $(1 - \theta_2) \rightarrow C'(\theta_2) = 2$

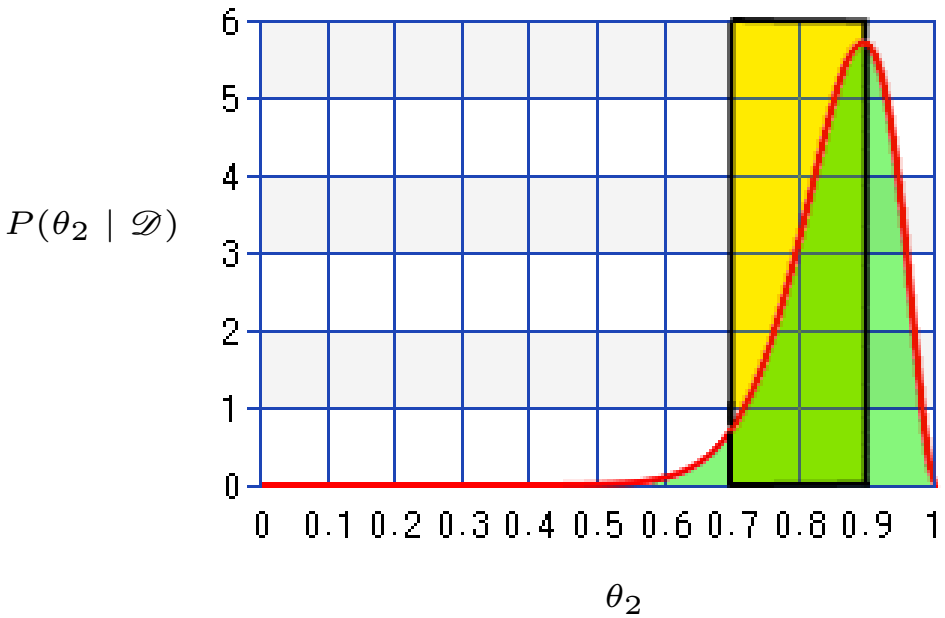
Overview


OBSERVATIONS
(LIMITED DATA)
BAYESIAN INFERENCE
SYSTEM
ACTIVE LEARNING
CONFIDENCE
COMPUTATION
PARTIAL MODEL
PARAMETER SYNTHESIS
PCTL PROPERTY

Confidence Computation

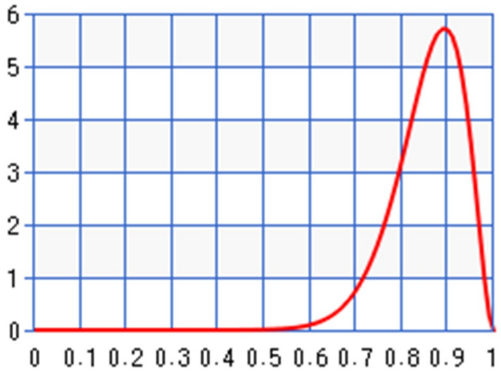
Confidence $\mathscr{C}$ on whether the system S satisfies the property φ:

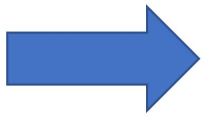
$$\mathscr{C} = \mathbb{P}(\mathsf{S} \models \varphi \mid \mathscr{D}) = \int_{\Theta_\varphi} P(\theta \mid \mathscr{D}) d\theta$$



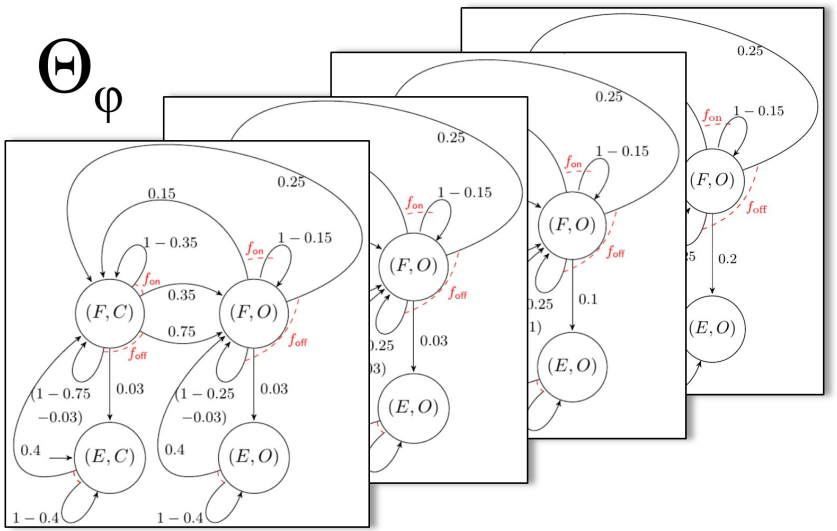
Confidence Computation

$P(\theta \mid D)$



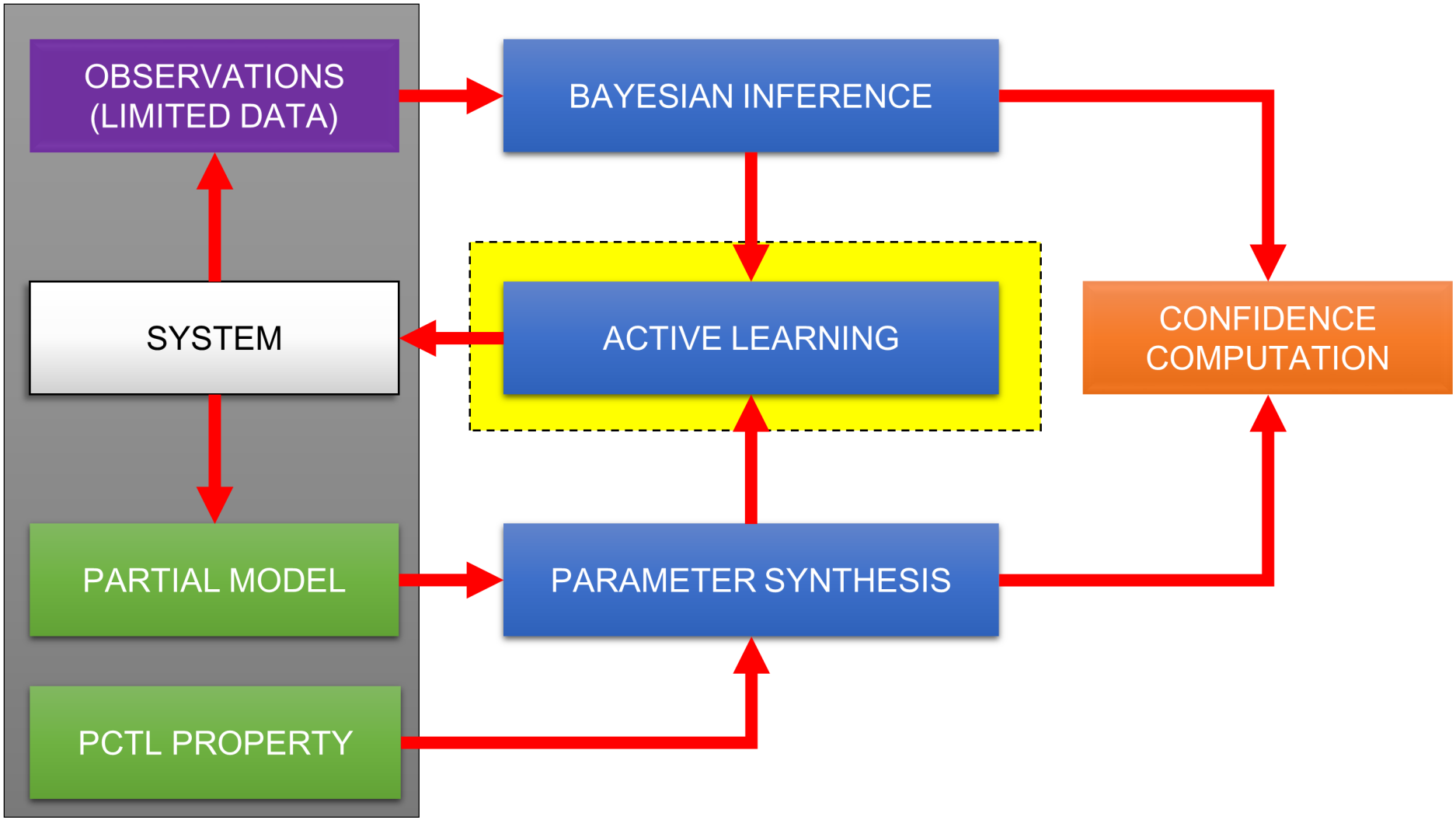


Confidence



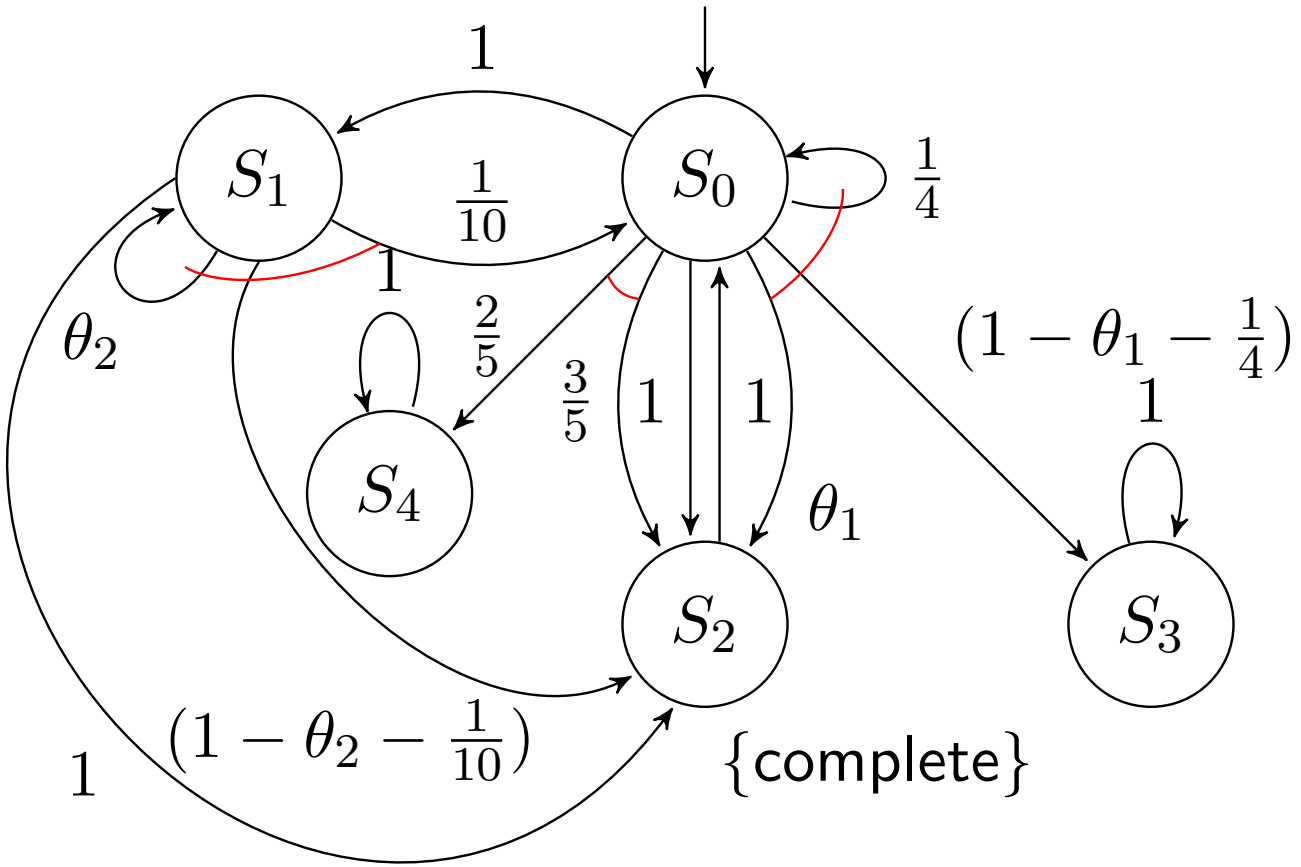
Outline

Overview


OBSERVATIONS
(LIMITED DATA)
BAYESIAN INFERENCE
SYSTEM
ACTIVE LEARNING
CONFIDENCE
COMPUTATION
PARTIAL MODEL
PARAMETER SYNTHESIS
PCTL PROPERTY

Data-Efficiency

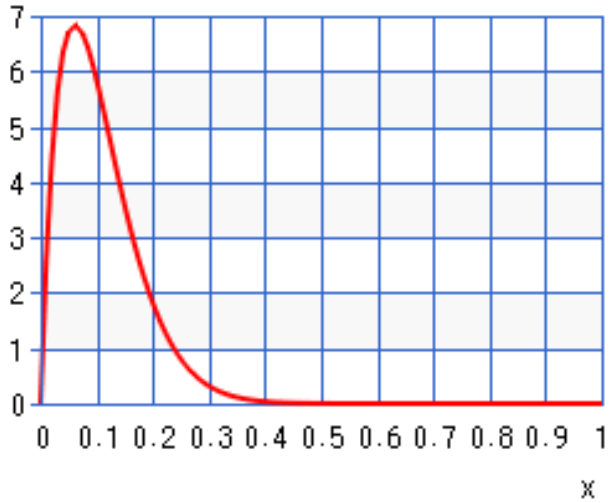
Some data tells us more about whether the property is satisfied



$\mathbf{P}_{\geq 0.5}(\text{true } \mathcal{U} \text{ complete})$

Role of Actions in Data Acquisition

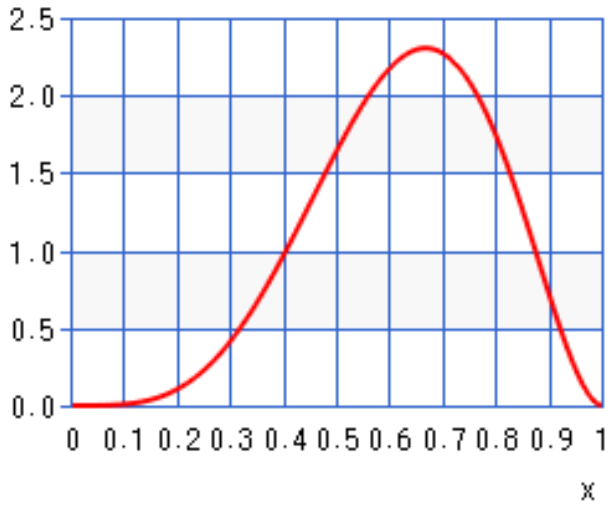
Actions can affect posterior distribution $P(\theta \mid \mathscr{D})$



Actions can affect confidence $\mathscr{C}$

Role of Actions in Data Acquisition

Actions can affect posterior distribution $P(\theta \mid \mathscr{D})$



Actions can affect confidence $\mathscr{C}$

Information State Space

Employs the joint space of states and information

Information State (Memory) $\mathbf{m} = s_0 a_0 s_1 a_1 s_2 a_2 \ldots$

Belief State $\hat{s} = (s, \mathbf{m})$

$\hat{s}$ is a statistic of the memory: $\hat{s} = f(\mathbf{m})$ and summarises all information accumulated so far

$$P(\theta_i \mid \mathscr{D}) = Beta(\theta_i; \bar{C}(\theta_i) + \omega) \propto \theta_i^{C(\theta_i)} (1 - \theta_i)^{C'(\theta_i)}$$

$$\mathbf{m} \leftarrow \mathrm{i}_{\mathbf{m}}$$

Joint Beta distribution with hyperparameters $(\boldsymbol{\alpha}, \boldsymbol{\beta})$

$$\boldsymbol{\alpha} = \langle C(\theta_1) + \alpha_{\theta_1}, C(\theta_2) + \alpha_{\theta_2}, \ldots, C(\theta_n) + \alpha_{\theta_n} \rangle$$

$$\boldsymbol{\beta} = \langle C'(\theta_1) + \beta_{\theta_1}, C'(\theta_2) + \beta_{\theta_2}, \ldots, C'(\theta_n) + \beta_{\theta_n} \rangle$$

$$(s, \mathbf{m}) \leftarrow (s, \mathrm{i}_{\mathbf{m}})$$

Information State Space

Employs the joint space of states and information

Information State (Memory) $\mathbf{m} = s_0 a_0 s_1 a_1 s_2 a_2 \ldots$

Belief State $\hat{s} = (s, \mathbf{m})$

$\hat{s}$ is a statistic of the memory: $\hat{s} = f(\mathbf{m})$ and summarises all information accumulated so far

$$P(\theta_i \mid \mathscr{D}) = Beta(\theta_i; \bar{C}(\theta_i) + \omega) \propto \theta_i^{C(\theta_i)} (1 - \theta_i)^{C'(\theta_i)}$$

$$\mathbf{m} \leftarrow \mathrm{i}_{\mathbf{m}}$$

Joint Beta distribution with hyperparameters $(\boldsymbol{\alpha}, \boldsymbol{\beta})$

$$\boldsymbol{\alpha} = \langle C(\theta_1) + \alpha_{\theta_1}, C(\theta_2) + \alpha_{\theta_2}, \ldots, C(\theta_n) + \alpha_{\theta_n} \rangle$$

$$\boldsymbol{\beta} = \langle C'(\theta_1) + \beta_{\theta_1}, C'(\theta_2) + \beta_{\theta_2}, \ldots, C'(\theta_n) + \beta_{\theta_n} \rangle$$

$$(s, \mathbf{m}) \leftarrow (s, \mathrm{i}_{\mathbf{m}})$$

Information State Space

Employs the joint space of states and information

Information State (Memory) $\mathbf{m} = s_0 a_0 s_1 a_1 s_2 a_2 \ldots$

Belief State $\hat{s} = (s, \mathbf{m})$

$\hat{s}$ is a statistic of the memory: $\hat{s} = f(\mathbf{m})$ and summarises all information accumulated so far

$$P(\theta_i \mid \mathscr{D}) = Beta(\theta_i; \bar{C}(\theta_i) + \omega) \propto \theta_i^{C(\theta_i)} (1 - \theta_i)^{C'(\theta_i)}$$

$\mathbf{m} \leftarrow \mathrm{i}_{\mathbf{m}}$

Joint Beta distribution with hyperparameters $(\boldsymbol{\alpha}, \boldsymbol{\beta})$

$\boldsymbol{\alpha} = \langle C(\theta_1) + \alpha_{\theta_1}, C(\theta_2) + \alpha_{\theta_2}, \ldots, C(\theta_n) + \alpha_{\theta_n} \rangle$

$\boldsymbol{\beta} = \langle C'(\theta_1) + \beta_{\theta_1}, C'(\theta_2) + \beta_{\theta_2}, \ldots, C'(\theta_n) + \beta_{\theta_n} \rangle$

$(s, \mathbf{m}) \leftarrow (s, \mathrm{i}_{\mathbf{m}})$

Information State Space

Employs the joint space of states and information

Information State (Memory) $\mathbf{m} = s_0 a_0 s_1 a_1 s_2 a_2 \ldots$

Belief State $\hat{s} = (s, \mathbf{m})$

$\hat{s}$ is a statistic of the memory: $\hat{s} = f(\mathbf{m})$ and summarises all information accumulated so far

$$P(\theta_i \mid \mathscr{D}) = Beta(\theta_i; \bar{C}(\theta_i) + \omega) \propto \theta_i^{C(\theta_i)} (1 - \theta_i)^{C'(\theta_i)}$$

$$\mathbf{m} \leftarrow \mathbf{i_m}$$

Joint Beta distribution with hyperparameters $(\boldsymbol{\alpha}, \boldsymbol{\beta})$

$$\boldsymbol{\alpha} = \langle C(\theta_1) + \alpha_{\theta_1}, C(\theta_2) + \alpha_{\theta_2}, \ldots, C(\theta_n) + \alpha_{\theta_n} \rangle$$

$$\boldsymbol{\beta} = \langle C'(\theta_1) + \beta_{\theta_1}, C'(\theta_2) + \beta_{\theta_2}, \ldots, C'(\theta_n) + \beta_{\theta_n} \rangle$$

$$(s, \mathbf{m}) \leftarrow (s, \mathbf{i_m})$$

Bayes-Adaptive MDP (BAMDP)

belief states — finite set of actions — $(s, \mathbf{m}) \in \hat{\iota}$ if $s \in \iota$

$$\mathcal{M}_{ba} = (\hat{S}, A, \hat{\mathcal{T}}, \hat{\iota}, \mathcal{R})$$

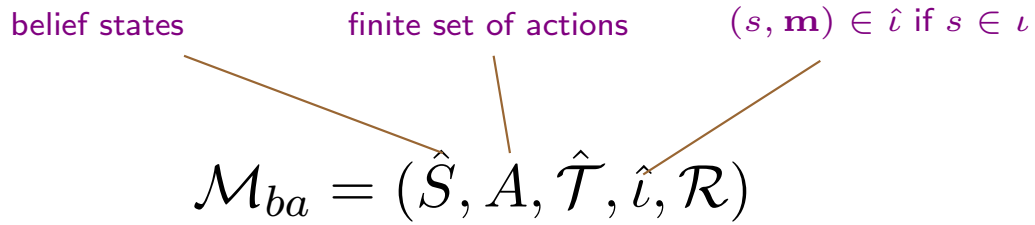
Extra information is the posterior distribution over the parameters

$$P(\theta \mid \mathscr{D}) = b(\mathcal{T}_p) = P(\mathcal{T}_p \mid \mathrm{i}_{\mathbf{m}}) \propto P(\mathrm{i}_{\mathbf{m}} \mid \mathcal{T}_p) P(\mathrm{i}_{\mathbf{m}})$$

$$\hat{\mathcal{T}}(s, \mathrm{i}_{\mathbf{m}}, a, s', \mathrm{i}_{\mathbf{m}'}) = \int_{\mathcal{T}_p} \mathcal{T}_p(s, a, s') P(\mathcal{T}_p \mid \mathrm{i}_{\mathbf{m}}) \mathrm{d}\mathcal{T}_p$$

$$\hat{\mathcal{T}}(s, \mathrm{i}_{\mathbf{m}}, a, s', \mathrm{i}_{\mathbf{m}'}) = \begin{cases} \mathbb{1}_{\mathrm{i}_{\mathbf{m}as'}}(\mathrm{i}_{\mathbf{m}'}) \frac{\alpha_{\theta_j}}{\alpha_{\theta_j} + \beta_{\theta_j}} & \text{if } \mathcal{T}_p(s, a, s') = \theta_j, \\ \mathbb{1}_{\mathrm{i}_{\mathbf{m}as'}}(\mathrm{i}_{\mathbf{m}'}) \frac{\beta_{\theta_j}}{\alpha_{\theta_j} + \beta_{\theta_j}} & \text{if } \mathcal{T}_p(s, a, s') \neq \theta_j \text{ and } \exists s_k \in S \colon \mathcal{T}_p(s, a, s_k) = \theta_j, \\ \mathcal{T}_p(s, a, s') & \text{otherwise} \end{cases}$$

Bayes-Adaptive MDP (BAMDP)



Extra information is the posterior distribution over the parameters

$$P(\theta \mid \mathscr{D}) = b(\mathcal{T}_p) = P(\mathcal{T}_p \mid \mathfrak{i}_{\mathbf{m}}) \propto P(\mathfrak{i}_{\mathbf{m}} \mid \mathcal{T}_p) P(\mathfrak{i}_{\mathbf{m}})$$

$$\hat{\mathcal{T}}(s, \mathfrak{i}_{\mathbf{m}}, a, s', \mathfrak{i}_{\mathbf{m}'}) = \int_{\mathcal{T}_p} \mathcal{T}_p(s, a, s') P(\mathcal{T}_p \mid \mathfrak{i}_{\mathbf{m}}) \mathrm{d}\mathcal{T}_p$$

$$\hat{\mathcal{T}}(s, \mathfrak{i}_{\mathbf{m}}, a, s', \mathfrak{i}_{\mathbf{m}'}) = \begin{cases} \mathbb{1}_{\mathfrak{i}_{\mathbf{m}as'}}(\mathfrak{i}_{\mathbf{m}'}) \frac{\alpha_{\theta_j}}{\alpha_{\theta_j} + \beta_{\theta_j}} & \text{if } \mathcal{T}_p(s, a, s') = \theta_j, \\ \mathbb{1}_{\mathfrak{i}_{\mathbf{m}as'}}(\mathfrak{i}_{\mathbf{m}'}) \frac{\beta_{\theta_j}}{\alpha_{\theta_j} + \beta_{\theta_j}} & \text{if } \mathcal{T}_p(s, a, s') \neq \theta_j \text{ and } \exists s_k \in S \colon \mathcal{T}_p(s, a, s_k) = \theta_j, \\ \mathcal{T}_p(s, a, s') & \text{otherwise} \end{cases}$$

Bayes-Adaptive MDP (BAMDP)

belief states — finite set of actions — $(s, \mathbf{m}) \in \hat{\iota}$ if $s \in \iota$

$$\mathcal{M}_{ba} = (\hat{S}, A, \hat{\mathcal{T}}, \hat{\iota}, \mathcal{R})$$

Extra information is the posterior distribution over the parameters

$$P(\theta \mid \mathscr{D}) = b(\mathcal{T}_p) = P(\mathcal{T}_p \mid \mathfrak{i}_{\mathbf{m}}) \propto P(\mathfrak{i}_{\mathbf{m}} \mid \mathcal{T}_p) P(\mathfrak{i}_{\mathbf{m}})$$

$$\hat{\mathcal{T}}(s, \mathfrak{i}_{\mathbf{m}}, a, s', \mathfrak{i}_{\mathbf{m}'}) = \int_{\mathcal{T}_p} \mathcal{T}_p(s, a, s') P(\mathcal{T}_p \mid \mathfrak{i}_{\mathbf{m}}) \mathrm{d}\mathcal{T}_p$$

$$\hat{\mathcal{T}}(s, \mathfrak{i}_{\mathbf{m}}, a, s', \mathfrak{i}_{\mathbf{m}'}) = \begin{cases} \mathbb{1}_{\mathfrak{i}_{\mathbf{m}as'}}(\mathfrak{i}_{\mathbf{m}'}) \frac{\alpha_{\theta_j}}{\alpha_{\theta_j} + \beta_{\theta_j}} & \text{if } \mathcal{T}_p(s, a, s') = \theta_j, \\ \mathbb{1}_{\mathfrak{i}_{\mathbf{m}as'}}(\mathfrak{i}_{\mathbf{m}'}) \frac{\beta_{\theta_j}}{\alpha_{\theta_j} + \beta_{\theta_j}} & \text{if } \mathcal{T}_p(s, a, s') \neq \theta_j \text{ and } \exists s_k \in S \colon \mathcal{T}_p(s, a, s_k) = \theta_j, \\ \mathcal{T}_p(s, a, s') & \text{otherwise} \end{cases}$$

Bayes-Adaptive MDP (BAMDP)

belief states — finite set of actions — $(s, \mathbf{m}) \in \hat{\iota}$ if $s \in \iota$

$$\mathcal{M}_{ba} = (\hat{S}, A, \hat{\mathcal{T}}, \hat{\iota}, \mathcal{R})$$

Extra information is the posterior distribution over the parameters

$$P(\theta \mid \mathscr{D}) = b(\mathcal{T}_p) = P(\mathcal{T}_p \mid \mathfrak{i}_{\mathbf{m}}) \propto P(\mathfrak{i}_{\mathbf{m}} \mid \mathcal{T}_p) P(\mathfrak{i}_{\mathbf{m}})$$

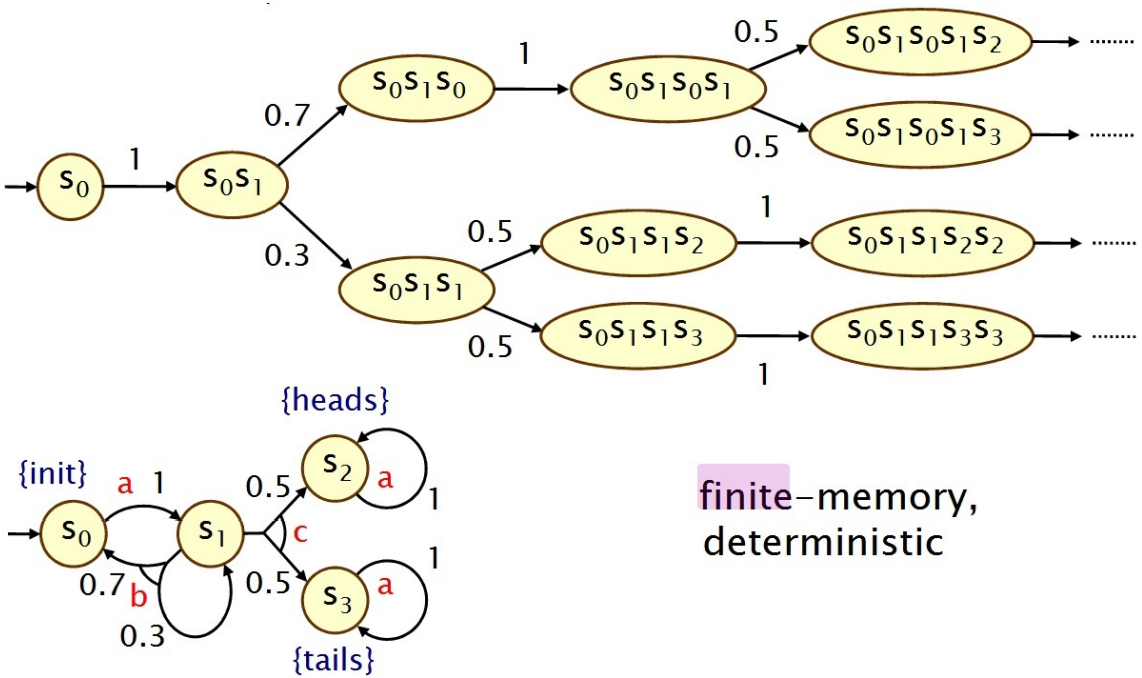
$$\hat{\mathcal{T}}(s, \mathfrak{i}_{\mathbf{m}}, a, s', \mathfrak{i}_{\mathbf{m}'}) = \int_{\mathcal{T}_p} \mathcal{T}_p(s, a, s') P(\mathcal{T}_p \mid \mathfrak{i}_{\mathbf{m}}) \mathrm{d}\mathcal{T}_p$$

$$\hat{\mathcal{T}}(s, \mathfrak{i}_{\mathbf{m}}, a, s', \mathfrak{i}_{\mathbf{m}'}) = \begin{cases} \mathbb{1}_{\mathfrak{i}_{\mathbf{m}as'}}(\mathfrak{i}_{\mathbf{m}'}) \frac{\alpha_{\theta_j}}{\alpha_{\theta_j} + \beta_{\theta_j}} & \text{if } \mathcal{T}_p(s, a, s') = \theta_j, \\ \mathbb{1}_{\mathfrak{i}_{\mathbf{m}as'}}(\mathfrak{i}_{\mathbf{m}'}) \frac{\beta_{\theta_j}}{\alpha_{\theta_j} + \beta_{\theta_j}} & \text{if } \mathcal{T}_p(s, a, s') \neq \theta_j \text{ and } \exists s_k \in S \colon \mathcal{T}_p(s, a, s_k) = \theta_j, \\ \mathcal{T}_p(s, a, s') & \text{otherwise} \end{cases}$$

Classification of Strategies

Memoryless strategy: $a_t = \pi(s_t), \forall t > 0$

Finite-memory strategy: There are finitely many modes M s.t. $a_t = \hat{\pi}(s_t, m_t), \forall t > 0$ where $m_t \in M$ is the current mode



Choose b then c in s_1

Figure courtesy of Marta Kwiatkowska

Reward Function

Data-Efficiency: want to maximise the information content of the observations

Need immediate rewards:- **immediate confidence gain**:

$$r_{t+1} = |0.5 - \mathscr{C}_{t+1}| - |0.5 - \mathscr{C}_t|$$

Want to maximise gain:- design reward function for BAMDP:

$$\mathcal{R}(s, \mathbf{i_m}, a) = \mathbb{E}[r_{t+1} \mid s_t = s, \mathbf{i}_{\mathbf{m}_t} = \mathbf{i_m}, a_t = a]$$

Q function: $Q^{\hat{\pi}}(s, \mathbf{m}, a) = \mathbb{E}_{\hat{\pi}}[\Sigma_{j=t}^{T} r_j \mid s_t = s, \mathbf{m}_t = \mathbf{m}, a_t = a]$

Bayes-Optimal strategy w.r.t prior $P(\theta)$:

$$\hat{\pi}^*(s, \mathbf{i_m}) = \arg\max_{a \in A(s)} Q^*(s, \mathbf{i_m}, a)$$

Reward Function

Data-Efficiency: want to maximise the information content of the observations

Need immediate rewards:- **immediate confidence gain**:

$$r_{t+1} = |0.5 - \mathscr{C}_{t+1}| - |0.5 - \mathscr{C}_t|$$

Want to maximise gain:- design reward function for BAMDP:

$$\mathcal{R}(s, \mathbf{i_m}, a) = \mathbb{E}[r_{t+1} \mid s_t = s, \mathbf{i}_{\mathbf{m}_t} = \mathbf{i_m}, a_t = a]$$

Q function: $Q^{\hat{\pi}}(s, \mathbf{m}, a) = \mathbb{E}_{\hat{\pi}}[\Sigma_{j=t}^{T} r_j \mid s_t = s, \mathbf{m}_t = \mathbf{m}, a_t = a]$

Bayes-Optimal strategy w.r.t prior $P(\theta)$:

$$\hat{\pi}^*(s, \mathbf{i_m}) = \arg\max_{a \in A(s)} Q^*(s, \mathbf{i_m}, a)$$

Reward Function

Data-Efficiency: want to maximise the information content of the observations

Need immediate rewards:- **immediate confidence gain**:

$$r_{t+1} = |0.5 - \mathscr{C}_{t+1}| - |0.5 - \mathscr{C}_t|$$

Want to maximise gain:- design reward function for BAMDP:

$$\mathcal{R}(s, \mathbf{i_m}, a) = \mathbb{E}[r_{t+1} \mid s_t = s, \mathbf{i}_{\mathbf{m}_t} = \mathbf{i_m}, a_t = a]$$

Q function: $Q^{\hat{\pi}}(s, \mathbf{m}, a) = \mathbb{E}_{\hat{\pi}}[\Sigma_{j=t}^{T} r_j \mid s_t = s, \mathbf{m}_t = \mathbf{m}, a_t = a]$

Bayes-Optimal strategy w.r.t prior $P(\theta)$:

$$\hat{\pi}^*(s, \mathbf{i_m}) = \arg\max_{a \in A(s)} Q^*(s, \mathbf{i_m}, a)$$

Reward Function

Data-Efficiency: want to maximise the information content of the observations

Need immediate rewards:- **immediate confidence gain**:

$$r_{t+1} = |0.5 - \mathscr{C}_{t+1}| - |0.5 - \mathscr{C}_t|$$

Want to maximise gain:- design reward function for BAMDP:

$$\mathcal{R}(s, \mathbf{i_m}, a) = \mathbb{E}[r_{t+1} \mid s_t = s, \mathbf{i}_{\mathbf{m}_t} = \mathbf{i_m}, a_t = a]$$

Q function: $Q^{\hat{\pi}}(s, \mathbf{m}, a) = \mathbb{E}_{\hat{\pi}}[\Sigma_{j=t}^{T} r_j \mid s_t = s, \mathbf{m}_t = \mathbf{m}, a_t = a]$

Bayes-Optimal strategy w.r.t prior $P(\theta)$:

$$\hat{\pi}^*(s, \mathbf{i_m}) = \arg\max_{a \in A(s)} Q^*(s, \mathbf{i_m}, a)$$

Reward Function

Data-Efficiency: want to maximise the information content of the observations

Need immediate rewards:- **immediate confidence gain**:

$$r_{t+1} = |0.5 - \mathscr{C}_{t+1}| - |0.5 - \mathscr{C}_t|$$

Want to maximise gain:- design reward function for BAMDP:

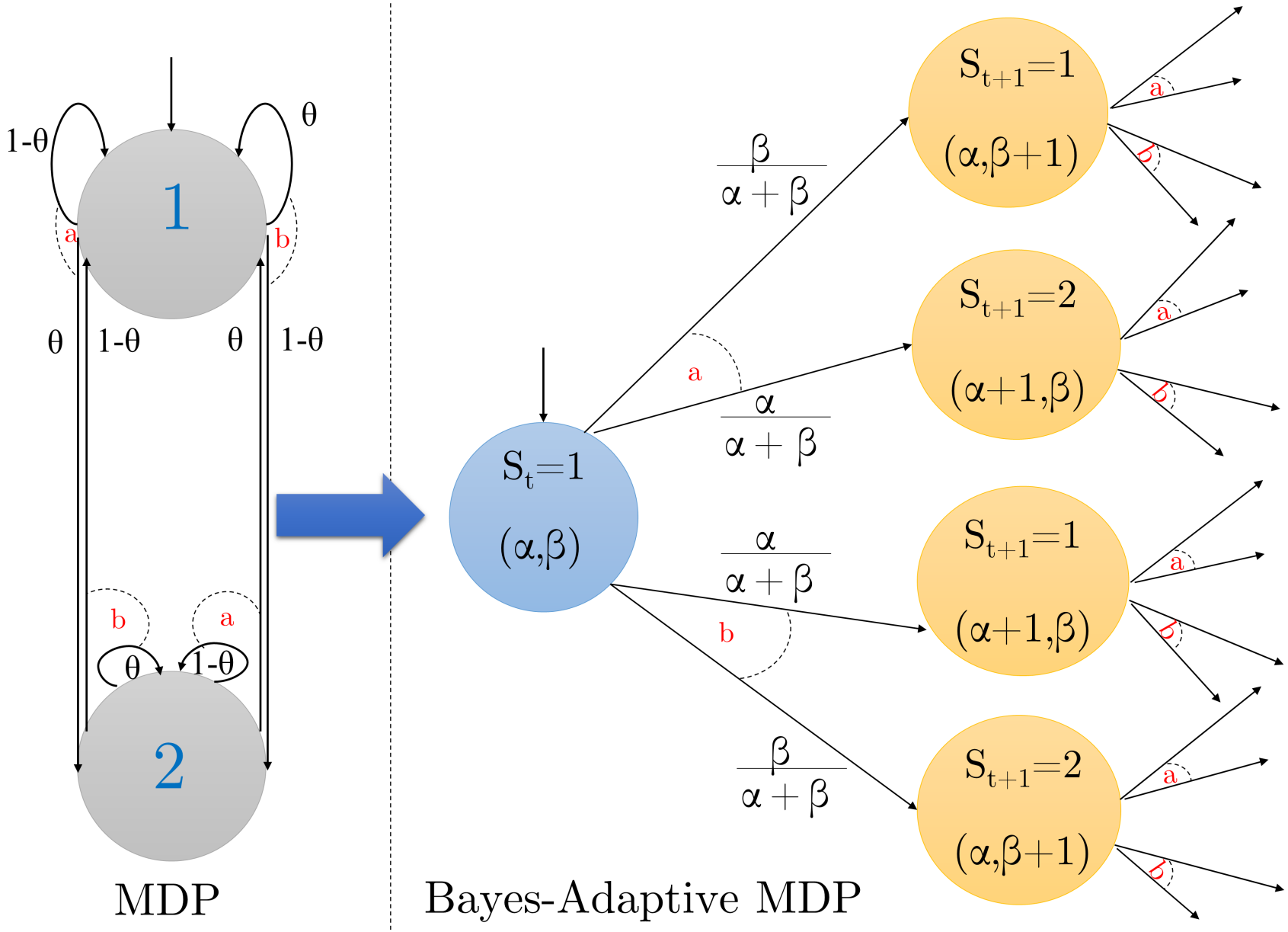
$$\mathcal{R}(s, \mathbf{i_m}, a) = \mathbb{E}[r_{t+1} \mid s_t = s, \mathbf{i}_{\mathbf{m}_t} = \mathbf{i_m}, a_t = a]$$

Q function: $Q^{\hat{\pi}}(s, \mathbf{m}, a) = \mathbb{E}_{\hat{\pi}}[\Sigma_{j=t}^{T} r_j \mid s_t = s, \mathbf{m}_t = \mathbf{m}, a_t = a]$

Bayes-Optimal strategy w.r.t prior $P(\theta)$:

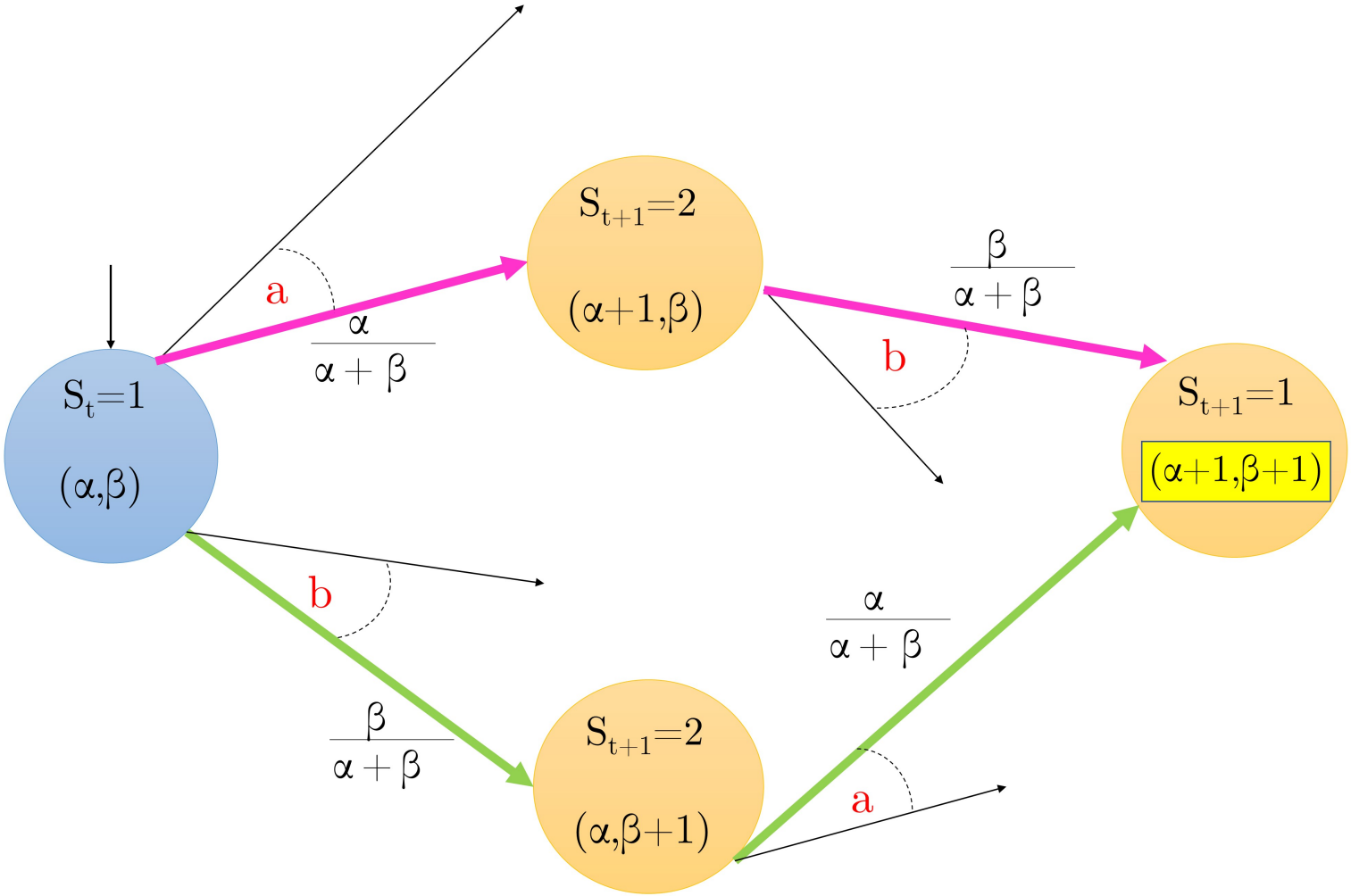
$$\hat{\pi}^*(s, \mathbf{i_m}) = \arg\max_{a \in A(s)} Q^*(s, \mathbf{i_m}, a)$$

Bayes-Adaptive MDPs are Sizeable



Value Function Approximation

Different memories can correspond to the same posterior belief



Generalise from seen to unseen states

Outline

Approximate Bilinear Q Function

$$Q(s, \mathbf{i_m}, a; \beta) = \mathbf{y}(\mathbf{i_m})^\top \beta \mathbf{x}(s, a)$$

- Goal of the learning process is to find β that minimises $\mathbb{E}[(Q(s, \mathbf{i_m}, a; \beta) - Q(s, \mathbf{i_m}, a))^2]$
- $\mathbf{x}(s, a)$ is a domain-dependent state-action feature vector: we use one-hot encoding to represent $\mathbf{x}(s, a)$
- Construct $\mathbf{y}\colon \widehat{\mathbf{M}} \to \mathbb{R}^{\mathcal{Z}}$ with $\mathcal{Z}$ as the degree of finite dimensional approximation and $\mathbf{i_m} \in \widehat{\mathbf{M}}$
- Update $\mathbf{y}_j(\mathbf{i_{mas'}}) = \mathbf{y}_j(\mathbf{i_m})\hat{\mathcal{T}}(s, \mathbf{i_m}, a, s', \mathbf{i_{mas'}})$ allows generalisation:

$$\mathbf{y}(\mathbf{i_{m'}}) = \mathbf{y}(\mathbf{i_m}) \text{ if } b(\mathbf{m'}) = b(\mathbf{m})$$

Approximate Bilinear Q Function

$$Q(s, \mathbf{i_m}, a; \beta) = \mathbf{y}(\mathbf{i_m})^\top \beta \mathbf{x}(s, a)$$

- Goal of the learning process is to find β that minimises $\mathbb{E}[(Q(s, \mathbf{i_m}, a; \beta) - Q(s, \mathbf{i_m}, a))^2]$
- $\mathbf{x}(s, a)$ is a domain-dependent state-action feature vector: we use one-hot encoding to represent $\mathbf{x}(s, a)$
- Construct $\mathbf{y}: \widehat{\mathbf{M}} \to \mathbb{R}^{\mathcal{Z}}$ with $\mathcal{Z}$ as the degree of finite dimensional approximation and $\mathbf{i_m} \in \widehat{\mathbf{M}}$
- Update $\mathbf{y}_j(\mathbf{i_{mas'}}) = \mathbf{y}_j(\mathbf{i_m})\hat{\mathcal{T}}(s, \mathbf{i_m}, a, s', \mathbf{i_{mas'}})$ allows generalisation:

$$\mathbf{y}(\mathbf{i_{m'}}) = \mathbf{y}(\mathbf{i_m}) \text{ if } b(\mathbf{m'}) = b(\mathbf{m})$$

Approximate Bilinear Q Function

$$Q(s, \mathbf{i_m}, a; \beta) = \mathbf{y}(\mathbf{i_m})^\top \beta \mathbf{x}(s, a)$$

- Goal of the learning process is to find β that minimises $\mathbb{E}[(Q(s, \mathbf{i_m}, a; \beta) - Q(s, \mathbf{i_m}, a))^2]$
- $\mathbf{x}(s, a)$ is a domain-dependent state-action feature vector: we use one-hot encoding to represent $\mathbf{x}(s, a)$
- Construct $\mathbf{y}$: $\widehat{\mathbf{M}} \to \mathbb{R}^{\mathcal{Z}}$ with $\mathcal{Z}$ as the degree of finite dimensional approximation and $\mathbf{i_m} \in \widehat{\mathbf{M}}$
- Update $\mathbf{y}_j(\mathbf{i_{mas'}}) = \mathbf{y}_j(\mathbf{i_m})\hat{\mathcal{T}}(s, \mathbf{i_m}, a, s', \mathbf{i_{mas'}})$ allows generalisation:

$$\mathbf{y}(\mathbf{i_{m'}}) = \mathbf{y}(\mathbf{i_m}) \text{ if } b(\mathbf{m'}) = b(\mathbf{m})$$

Approximate Bilinear Q Function

$$Q(s, \mathbf{i_m}, a; \beta) = \mathbf{y}(\mathbf{i_m})^\top \beta \mathbf{x}(s, a)$$

- Goal of the learning process is to find β that minimises $\mathbb{E}[(Q(s, \mathbf{i_m}, a; \beta) - Q(s, \mathbf{i_m}, a))^2]$
- $\mathbf{x}(s, a)$ is a domain-dependent state-action feature vector: we use one-hot encoding to represent $\mathbf{x}(s, a)$
- Construct $\mathbf{y}$: $\widehat{\mathbf{M}} \to \mathbb{R}^{\mathcal{Z}}$ with $\mathcal{Z}$ as the degree of finite dimensional approximation and $\mathbf{i_m} \in \widehat{\mathbf{M}}$
- Update $\mathbf{y}_j(\mathbf{i_{mas'}}) = \mathbf{y}_j(\mathbf{i_m})\hat{\mathcal{T}}(s, \mathbf{i_m}, a, s', \mathbf{i_{mas'}})$ allows generalisation:

$$\mathbf{y}(\mathbf{i_{m'}}) = \mathbf{y}(\mathbf{i_m}) \text{ if } b(\mathbf{m'}) = b(\mathbf{m})$$

Bayes-Adaptive Temporal Difference Search

Inputs:
  s_t, $\mathbf{i}_{\mathbf{m}_t}$
Initialize:
  $\beta \leftarrow 0$, $\rho \leftarrow 0$
procedure SEARCH(s_t, $\mathbf{i}_{\mathbf{m}_t}$)
  while time remaining **do** ▷ Start episode
    $s \leftarrow s_t$, $\mathbf{i}_{\mathbf{m}} \leftarrow \mathbf{i}_{\mathbf{m}_t}$, $\tilde{t} \leftarrow t$
    $a \leftarrow \hat{\pi}_{\epsilon\text{–greedy}}(s, \mathbf{i}_{\mathbf{m}}; Q)$
    $\xi \leftarrow 0$, $\xi' \leftarrow 0$
    while $\tilde{t} < T$ **do**
      $s' \sim \hat{\mathcal{T}}(s, \mathbf{i}_{\mathbf{m}}, a, \cdot, \mathbf{i}_{\mathbf{ma}\cdot})$
      $R \leftarrow \mathcal{R}(s, \mathbf{i}_{\mathbf{m}}, a)$
      $a' \leftarrow \hat{\pi}_{\epsilon\text{–greedy}}(s', \mathbf{i}_{\mathbf{m}as'}; Q)$
      $\delta \leftarrow R + Q(s', \mathbf{i}_{\mathbf{m}as'}, a'; \beta) - Q(s, \mathbf{i}_{\mathbf{m}}, a; \beta)$
      $\xi = \lambda\xi + \mathbf{y}(\mathbf{i}_{\mathbf{m}}) \otimes \mathbf{x}(s, a)$
      $\xi' = \lambda\xi + \mathbf{y}(\mathbf{i}_{\mathbf{m}as'}) \otimes \mathbf{x}(s', a')$
      $\beta \leftarrow \beta + \alpha\delta\xi - \alpha\xi'(\xi^\top\rho)$
      $\rho \leftarrow \rho + \omega(\delta J - \xi^\top\rho)\xi$
      $s \leftarrow s'$, $a \leftarrow a'$, $\tilde{t} \leftarrow \tilde{t} + 1$
    end while
  end while
  return $\arg\max_{a_t} Q(s_t, \mathbf{i}_{\mathbf{m}_t}, a_t; \beta)$
end procedure

Bayes-Adaptive Temporal Difference Search

Inputs:
  s_t, $\mathbf{i}_{\mathbf{m}_t}$
Initialize:
  $\beta \leftarrow 0$, $\rho \leftarrow 0$
procedure SEARCH(s_t, $\mathbf{i}_{\mathbf{m}_t}$)
  while time remaining **do** ▷ Start episode
    $s \leftarrow s_t$, $\mathbf{i}_{\mathbf{m}} \leftarrow \mathbf{i}_{\mathbf{m}_t}$, $\tilde{t} \leftarrow t$
    $a \leftarrow \hat{\pi}_{\epsilon\text{–greedy}}(s, \mathbf{i}_{\mathbf{m}}; Q)$
    $\xi \leftarrow 0$, $\xi' \leftarrow 0$
    while $\tilde{t} < T$ **do**
      $s' \sim \hat{\mathcal{T}}(s, \mathbf{i}_{\mathbf{m}}, a, \cdot, \mathbf{i}_{\mathbf{ma}\cdot})$
      $R \leftarrow \mathcal{R}(s, \mathbf{i}_{\mathbf{m}}, a)$
      $a' \leftarrow \hat{\pi}_{\epsilon\text{–greedy}}(s', \mathbf{i}_{\mathbf{m}as'}; Q)$
      $\delta \leftarrow R + Q(s', \mathbf{i}_{\mathbf{m}as'}, a'; \beta) - Q(s, \mathbf{i}_{\mathbf{m}}, a; \beta)$
      $\xi = \lambda\xi + \mathbf{y}(\mathbf{i}_{\mathbf{m}}) \otimes \mathbf{x}(s, a)$
      $\xi' = \lambda\xi + \mathbf{y}(\mathbf{i}_{\mathbf{m}as'}) \otimes \mathbf{x}(s', a')$
      $\beta \leftarrow \beta + \alpha\delta\xi - \alpha\xi'(\xi^\top\rho)$
      $\rho \leftarrow \rho + \omega(\delta J - \xi^\top\rho)\xi$
      $s \leftarrow s'$, $a \leftarrow a'$, $\tilde{t} \leftarrow \tilde{t} + 1$
    end while
  end while
  return $\arg\max_{a_t} Q(s_t, \mathbf{i}_{\mathbf{m}_t}, a_t; \beta)$
end procedure

Bayes-Adaptive Temporal Difference Search

Inputs:
  s_t, $\mathbf{i}_{\mathbf{m}_t}$
Initialize:
  $\beta \leftarrow 0$, $\rho \leftarrow 0$
procedure SEARCH$(s_t, \mathbf{i}_{\mathbf{m}_t})$
  while time remaining **do** ▷ Start episode
    $s \leftarrow s_t$, $\mathbf{i}_{\mathbf{m}} \leftarrow \mathbf{i}_{\mathbf{m}_t}$, $\tilde{t} \leftarrow t$
    $a \leftarrow \hat{\pi}_{\epsilon\text{–greedy}}(s, \mathbf{i}_{\mathbf{m}}; Q)$
    $\xi \leftarrow 0$, $\xi' \leftarrow 0$
    while $\tilde{t} < T$ **do**
      $s' \sim \hat{\mathcal{T}}(s, \mathbf{i}_{\mathbf{m}}, a, \cdot, \mathbf{i}_{\mathbf{ma}\cdot})$
      $R \leftarrow \mathcal{R}(s, \mathbf{i}_{\mathbf{m}}, a)$
      $a' \leftarrow \hat{\pi}_{\epsilon\text{–greedy}}(s', \mathbf{i}_{\mathbf{m}as'}; Q)$
      $\delta \leftarrow R + Q(s', \mathbf{i}_{\mathbf{m}as'}, a'; \beta) - Q(s, \mathbf{i}_{\mathbf{m}}, a; \beta)$
      $\xi = \lambda\xi + \mathbf{y}(\mathbf{i}_{\mathbf{m}}) \otimes \mathbf{x}(s, a)$
      $\xi' = \lambda\xi + \mathbf{y}(\mathbf{i}_{\mathbf{m}as'}) \otimes \mathbf{x}(s', a')$
      $\beta \leftarrow \beta + \alpha\delta\xi - \alpha\xi'(\xi^\top\rho)$
      $\rho \leftarrow \rho + \omega(\delta J - \xi^\top\rho)\xi$
      $s \leftarrow s'$, $a \leftarrow a'$, $\tilde{t} \leftarrow \tilde{t} + 1$
    end while
  end while
  return $\arg\max_{a_t} Q(s_t, \mathbf{i}_{\mathbf{m}_t}, a_t; \beta)$
end procedure

Bayes-Adaptive Temporal Difference Search

Inputs:
  s_t, $\mathbf{i}_{\mathbf{m}_t}$
Initialize:
  $\beta \leftarrow 0$, $\rho \leftarrow 0$
procedure SEARCH(s_t, $\mathbf{i}_{\mathbf{m}_t}$)
  while time remaining **do** ▷ Start episode
    $s \leftarrow s_t$, $\mathbf{i}_{\mathbf{m}} \leftarrow \mathbf{i}_{\mathbf{m}_t}$, $\tilde{t} \leftarrow t$
    $a \leftarrow \hat{\pi}_{\epsilon\text{–greedy}}(s, \mathbf{i}_{\mathbf{m}}; Q)$
    $\xi \leftarrow 0$, $\xi' \leftarrow 0$
    while $\tilde{t} < T$ **do**
      $s' \sim \hat{\mathcal{T}}(s, \mathbf{i}_{\mathbf{m}}, a, \cdot, \mathbf{i}_{\mathbf{ma}\cdot})$
      $R \leftarrow \mathcal{R}(s, \mathbf{i}_{\mathbf{m}}, a)$
      $a' \leftarrow \hat{\pi}_{\epsilon\text{–greedy}}(s', \mathbf{i}_{\mathbf{m}as'}; Q)$
      $\delta \leftarrow R + Q(s', \mathbf{i}_{\mathbf{m}as'}, a'; \beta) - Q(s, \mathbf{i}_{\mathbf{m}}, a; \beta)$
      $\xi = \lambda\xi + \mathbf{y}(\mathbf{i}_{\mathbf{m}}) \otimes \mathbf{x}(s, a)$
      $\xi' = \lambda\xi + \mathbf{y}(\mathbf{i}_{\mathbf{m}as'}) \otimes \mathbf{x}(s', a')$
      $\beta \leftarrow \beta + \alpha\delta\xi - \alpha\xi'(\xi^\top\rho)$
      $\rho \leftarrow \rho + \omega(\delta J - \xi^\top\rho)\xi$
      $s \leftarrow s'$, $a \leftarrow a'$, $\tilde{t} \leftarrow \tilde{t} + 1$
    end while
  end while
  return $\arg\max_{a_t} Q(s_t, \mathbf{i}_{\mathbf{m}_t}, a_t; \beta)$
end procedure

Bayes-Adaptive Temporal Difference Search

Inputs:
  s_t, $\mathbf{i}_{\mathbf{m}_t}$
Initialize:
  $\beta \leftarrow 0$, $\rho \leftarrow 0$
procedure SEARCH(s_t, $\mathbf{i}_{\mathbf{m}_t}$)
  while time remaining **do** ▷ Start episode
    $s \leftarrow s_t$, $\mathbf{i}_{\mathbf{m}} \leftarrow \mathbf{i}_{\mathbf{m}_t}$, $\tilde{t} \leftarrow t$
    $a \leftarrow \hat{\pi}_{\epsilon\text{–greedy}}(s, \mathbf{i}_{\mathbf{m}}; Q)$
    $\xi \leftarrow 0$, $\xi' \leftarrow 0$
    while $\tilde{t} < T$ **do**
      $s' \sim \hat{\mathcal{T}}(s, \mathbf{i}_{\mathbf{m}}, a, \cdot, \mathbf{i}_{\mathbf{ma}\cdot})$
      $R \leftarrow \mathcal{R}(s, \mathbf{i}_{\mathbf{m}}, a)$
      $a' \leftarrow \hat{\pi}_{\epsilon\text{–greedy}}(s', \mathbf{i}_{\mathbf{m}as'}; Q)$
      $\delta \leftarrow R + Q(s', \mathbf{i}_{\mathbf{m}as'}, a'; \beta) - Q(s, \mathbf{i}_{\mathbf{m}}, a; \beta)$
      $\xi = \lambda\xi + \mathbf{y}(\mathbf{i}_{\mathbf{m}}) \otimes \mathbf{x}(s, a)$
      $\xi' = \lambda\xi + \mathbf{y}(\mathbf{i}_{\mathbf{m}as'}) \otimes \mathbf{x}(s', a')$
      $\beta \leftarrow \beta + \alpha\delta\xi - \alpha\xi'(\xi^\top\rho)$
      $\rho \leftarrow \rho + \omega(\delta J - \xi^\top\rho)\xi$
      $s \leftarrow s'$, $a \leftarrow a'$, $\tilde{t} \leftarrow \tilde{t} + 1$
    end while
  end while
  return $\arg\max_{a_t} Q(s_t, \mathbf{i}_{\mathbf{m}_t}, a_t; \beta)$
end procedure

Bayes-Adaptive Temporal Difference Search

Inputs:
  s_t, $\mathbf{i}_{\mathbf{m}_t}$
Initialize:
  $\beta \leftarrow 0$, $\rho \leftarrow 0$
procedure SEARCH(s_t, $\mathbf{i}_{\mathbf{m}_t}$)
  while time remaining **do** ▷ Start episode
    $s \leftarrow s_t$, $\mathbf{i}_{\mathbf{m}} \leftarrow \mathbf{i}_{\mathbf{m}_t}$, $\tilde{t} \leftarrow t$
    $a \leftarrow \hat{\pi}_{\epsilon\text{–greedy}}(s, \mathbf{i}_{\mathbf{m}}; Q)$
    $\xi \leftarrow 0$, $\xi' \leftarrow 0$
    while $\tilde{t} < T$ **do**
      $s' \sim \hat{\mathcal{T}}(s, \mathbf{i}_{\mathbf{m}}, a, \cdot, \mathbf{i}_{\mathbf{m}a\cdot})$
      $R \leftarrow \mathcal{R}(s, \mathbf{i}_{\mathbf{m}}, a)$
      $a' \leftarrow \hat{\pi}_{\epsilon\text{–greedy}}(s', \mathbf{i}_{\mathbf{m}as'}; Q)$
      $\delta \leftarrow R + Q(s', \mathbf{i}_{\mathbf{m}as'}, a'; \beta) - Q(s, \mathbf{i}_{\mathbf{m}}, a; \beta)$
      $\xi = \lambda\xi + \mathbf{y}(\mathbf{i}_{\mathbf{m}}) \otimes \mathbf{x}(s, a)$
      $\xi' = \lambda\xi + \mathbf{y}(\mathbf{i}_{\mathbf{m}as'}) \otimes \mathbf{x}(s', a')$
      $\beta \leftarrow \beta + \alpha\delta\xi - \alpha\xi'(\xi^\top\rho)$
      $\rho \leftarrow \rho + \omega(\delta J - \xi^\top\rho)\xi$
      $s \leftarrow s'$, $a \leftarrow a'$, $\tilde{t} \leftarrow \tilde{t} + 1$
    end while
  end while
  return $\arg\max_{a_t} Q(s_t, \mathbf{i}_{\mathbf{m}_t}, a_t; \beta)$
end procedure

Bayes-Adaptive Temporal Difference Search

Inputs:
  s_t, $\mathbf{i}_{\mathbf{m}_t}$
Initialize:
  $\beta \leftarrow 0$, $\rho \leftarrow 0$
procedure SEARCH(s_t, $\mathbf{i}_{\mathbf{m}_t}$)
  while time remaining **do** ▷ Start episode
    $s \leftarrow s_t$, $\mathbf{i}_{\mathbf{m}} \leftarrow \mathbf{i}_{\mathbf{m}_t}$, $\tilde{t} \leftarrow t$
    $a \leftarrow \hat{\pi}_{\epsilon\text{–greedy}}(s, \mathbf{i}_{\mathbf{m}}; Q)$
    $\xi \leftarrow 0$, $\xi' \leftarrow 0$
    while $\tilde{t} < T$ **do**
      $s' \sim \hat{\mathcal{T}}(s, \mathbf{i}_{\mathbf{m}}, a, \cdot, \mathbf{i}_{\mathbf{ma}\cdot})$
      $R \leftarrow \mathcal{R}(s, \mathbf{i}_{\mathbf{m}}, a)$
      $a' \leftarrow \hat{\pi}_{\epsilon\text{–greedy}}(s', \mathbf{i}_{\mathbf{m}as'}; Q)$
      $\delta \leftarrow R + Q(s', \mathbf{i}_{\mathbf{m}as'}, a'; \beta) - Q(s, \mathbf{i}_{\mathbf{m}}, a; \beta)$
      $\xi = \lambda\xi + \mathbf{y}(\mathbf{i}_{\mathbf{m}}) \otimes \mathbf{x}(s, a)$
      $\xi' = \lambda\xi + \mathbf{y}(\mathbf{i}_{\mathbf{m}as'}) \otimes \mathbf{x}(s', a')$
      $\beta \leftarrow \beta + \alpha\delta\xi - \alpha\xi'(\xi^\top\rho)$
      $\rho \leftarrow \rho + \omega(\delta J - \xi^\top\rho)\xi$
      $s \leftarrow s'$, $a \leftarrow a'$, $\tilde{t} \leftarrow \tilde{t} + 1$
    end while
  end while
  return $\arg\max_{a_t} Q(s_t, \mathbf{i}_{\mathbf{m}_t}, a_t; \beta)$
end procedure

Bayes-Adaptive Temporal Difference Search

Inputs:
  s_t, $\mathbf{i}_{\mathbf{m}_t}$
Initialize:
  $\beta \leftarrow 0$, $\rho \leftarrow 0$
procedure SEARCH(s_t, $\mathbf{i}_{\mathbf{m}_t}$)
  while time remaining **do** ▷ Start episode
    $s \leftarrow s_t$, $\mathbf{i}_{\mathbf{m}} \leftarrow \mathbf{i}_{\mathbf{m}_t}$, $\tilde{t} \leftarrow t$
    $a \leftarrow \hat{\pi}_{\epsilon\text{–greedy}}(s, \mathbf{i}_{\mathbf{m}}; Q)$
    $\xi \leftarrow 0$, $\xi' \leftarrow 0$
    while $\tilde{t} < T$ **do**
      $s' \sim \hat{\mathcal{T}}(s, \mathbf{i}_{\mathbf{m}}, a, \cdot, \mathbf{i}_{\mathbf{ma}\cdot})$
      $R \leftarrow \mathcal{R}(s, \mathbf{i}_{\mathbf{m}}, a)$
      $a' \leftarrow \hat{\pi}_{\epsilon\text{–greedy}}(s', \mathbf{i}_{\mathbf{m}as'}; Q)$
      $\delta \leftarrow R + Q(s', \mathbf{i}_{\mathbf{m}as'}, a'; \beta) - Q(s, \mathbf{i}_{\mathbf{m}}, a; \beta)$
      $\xi = \lambda\xi + \mathbf{y}(\mathbf{i}_{\mathbf{m}}) \otimes \mathbf{x}(s, a)$
      $\xi' = \lambda\xi + \mathbf{y}(\mathbf{i}_{\mathbf{m}as'}) \otimes \mathbf{x}(s', a')$
      $\beta \leftarrow \beta + \alpha\delta\xi - \alpha\xi'(\xi^\top\rho)$
      $\rho \leftarrow \rho + \omega(\delta J - \xi^\top\rho)\xi$
      $s \leftarrow s'$, $a \leftarrow a'$, $\tilde{t} \leftarrow \tilde{t} + 1$
    end while
  end while
  return $\arg\max_{a_t} Q(s_t, \mathbf{i}_{\mathbf{m}_t}, a_t; \beta)$
end procedure

Bayes-Adaptive Temporal Difference Search

Inputs:
  s_t, $\mathbf{i}_{\mathbf{m}_t}$
Initialize:
  $\beta \leftarrow 0$, $\rho \leftarrow 0$
procedure SEARCH(s_t, $\mathbf{i}_{\mathbf{m}_t}$)
  while time remaining **do** ▷ Start episode
    $s \leftarrow s_t$, $\mathbf{i}_{\mathbf{m}} \leftarrow \mathbf{i}_{\mathbf{m}_t}$, $\tilde{t} \leftarrow t$
    $a \leftarrow \hat{\pi}_{\epsilon\text{–greedy}}(s, \mathbf{i}_{\mathbf{m}}; Q)$
    $\xi \leftarrow 0$, $\xi' \leftarrow 0$
    while $\tilde{t} < T$ **do**
      $s' \sim \hat{\mathcal{T}}(s, \mathbf{i}_{\mathbf{m}}, a, \cdot, \mathbf{i}_{\mathbf{m}a\cdot})$
      $R \leftarrow \mathcal{R}(s, \mathbf{i}_{\mathbf{m}}, a)$
      $a' \leftarrow \hat{\pi}_{\epsilon\text{–greedy}}(s', \mathbf{i}_{\mathbf{m}as'}; Q)$
      $\delta \leftarrow R + Q(s', \mathbf{i}_{\mathbf{m}as'}, a'; \beta) - Q(s, \mathbf{i}_{\mathbf{m}}, a; \beta)$
      $\xi = \lambda\xi + \mathbf{y}(\mathbf{i}_{\mathbf{m}}) \otimes \mathbf{x}(s, a)$
      $\xi' = \lambda\xi + \mathbf{y}(\mathbf{i}_{\mathbf{m}as'}) \otimes \mathbf{x}(s', a')$
      $\beta \leftarrow \beta + \alpha\delta\xi - \alpha\xi'(\xi^\top\rho)$
      $\rho \leftarrow \rho + \omega(\delta J - \xi^\top\rho)\xi$
      $s \leftarrow s'$, $a \leftarrow a'$, $\tilde{t} \leftarrow \tilde{t} + 1$
    end while
  end while
  return $\arg\max_{a_t} Q(s_t, \mathbf{i}_{\mathbf{m}_t}, a_t; \beta)$
end procedure

Bayes-Adaptive Temporal Difference Search

Inputs:
  s_t, $\mathbf{i}_{\mathbf{m}_t}$
Initialize:
  $\beta \leftarrow 0$, $\rho \leftarrow 0$
procedure SEARCH(s_t, $\mathbf{i}_{\mathbf{m}_t}$)
  while time remaining **do** ▷ Start episode
    $s \leftarrow s_t$, $\mathbf{i}_{\mathbf{m}} \leftarrow \mathbf{i}_{\mathbf{m}_t}$, $\tilde{t} \leftarrow t$
    $a \leftarrow \hat{\pi}_{\epsilon\text{–greedy}}(s, \mathbf{i}_{\mathbf{m}}; Q)$
    $\xi \leftarrow 0$, $\xi' \leftarrow 0$
    while $\tilde{t} < T$ **do**
      $s' \sim \hat{\mathcal{T}}(s, \mathbf{i}_{\mathbf{m}}, a, \cdot, \mathbf{i}_{\mathbf{ma}\cdot})$
      $R \leftarrow \mathcal{R}(s, \mathbf{i}_{\mathbf{m}}, a)$
      $a' \leftarrow \hat{\pi}_{\epsilon\text{–greedy}}(s', \mathbf{i}_{\mathbf{m}as'}; Q)$
      $\delta \leftarrow R + Q(s', \mathbf{i}_{\mathbf{m}as'}, a'; \beta) - Q(s, \mathbf{i}_{\mathbf{m}}, a; \beta)$
      $\xi = \lambda\xi + \mathbf{y}(\mathbf{i}_{\mathbf{m}}) \otimes \mathbf{x}(s, a)$
      $\xi' = \lambda\xi + \mathbf{y}(\mathbf{i}_{\mathbf{m}as'}) \otimes \mathbf{x}(s', a')$
      $\beta \leftarrow \beta + \alpha\delta\xi - \alpha\xi'(\xi^\top\rho)$
      $\rho \leftarrow \rho + \omega(\delta J - \xi^\top\rho)\xi$
      $s \leftarrow s'$, $a \leftarrow a'$, $\tilde{t} \leftarrow \tilde{t} + 1$
    end while
  end while
  return $\arg\max_{a_t} Q(s_t, \mathbf{i}_{\mathbf{m}_t}, a_t; \beta)$
end procedure

Bayes-Adaptive Temporal Difference Search

Inputs:
  s_t, $\mathbf{i}_{\mathbf{m}_t}$
Initialize:
  $\beta \leftarrow 0$, $\rho \leftarrow 0$
procedure SEARCH(s_t, $\mathbf{i}_{\mathbf{m}_t}$)
  while time remaining **do** ▷ Start episode
    $s \leftarrow s_t$, $\mathbf{i}_{\mathbf{m}} \leftarrow \mathbf{i}_{\mathbf{m}_t}$, $\tilde{t} \leftarrow t$
    $a \leftarrow \hat{\pi}_{\epsilon\text{–greedy}}(s, \mathbf{i}_{\mathbf{m}}; Q)$
    $\xi \leftarrow 0$, $\xi' \leftarrow 0$
    while $\tilde{t} < T$ **do**
      $s' \sim \hat{\mathcal{T}}(s, \mathbf{i}_{\mathbf{m}}, a, \cdot, \mathbf{i}_{\mathbf{ma}\cdot})$
      $R \leftarrow \mathcal{R}(s, \mathbf{i}_{\mathbf{m}}, a)$
      $a' \leftarrow \hat{\pi}_{\epsilon\text{–greedy}}(s', \mathbf{i}_{\mathbf{m}as'}; Q)$
      $\delta \leftarrow R + Q(s', \mathbf{i}_{\mathbf{m}as'}, a'; \beta) - Q(s, \mathbf{i}_{\mathbf{m}}, a; \beta)$
      $\xi = \lambda\xi + \mathbf{y}(\mathbf{i}_{\mathbf{m}}) \otimes \mathbf{x}(s, a)$
      $\xi' = \lambda\xi + \mathbf{y}(\mathbf{i}_{\mathbf{m}as'}) \otimes \mathbf{x}(s', a')$
      $\beta \leftarrow \beta + \alpha\delta\xi - \alpha\xi'(\xi^\top\rho)$
      $\rho \leftarrow \rho + \omega(\delta J - \xi^\top\rho)\xi$
      $s \leftarrow s'$, $a \leftarrow a'$, $\tilde{t} \leftarrow \tilde{t} + 1$
    end while
  end while
  return $\arg\max_{a_t} Q(s_t, \mathbf{i}_{\mathbf{m}_t}, a_t; \beta)$
end procedure

Bayes-Adaptive Temporal Difference Search

Inputs:
  s_t, $\mathbf{i}_{\mathbf{m}_t}$
Initialize:
  $\beta \leftarrow 0$, $\rho \leftarrow 0$
procedure SEARCH(s_t, $\mathbf{i}_{\mathbf{m}_t}$)
  while time remaining **do** ▷ Start episode
    $s \leftarrow s_t$, $\mathbf{i}_{\mathbf{m}} \leftarrow \mathbf{i}_{\mathbf{m}_t}$, $\tilde{t} \leftarrow t$
    $a \leftarrow \hat{\pi}_{\epsilon\text{–greedy}}(s, \mathbf{i}_{\mathbf{m}}; Q)$
    $\xi \leftarrow 0$, $\xi' \leftarrow 0$
    while $\tilde{t} < T$ **do**
      $s' \sim \hat{\mathcal{T}}(s, \mathbf{i}_{\mathbf{m}}, a, \cdot, \mathbf{i}_{\mathbf{ma}\cdot})$
      $R \leftarrow \mathcal{R}(s, \mathbf{i}_{\mathbf{m}}, a)$
      $a' \leftarrow \hat{\pi}_{\epsilon\text{–greedy}}(s', \mathbf{i}_{\mathbf{m}as'}; Q)$
      $\delta \leftarrow R + Q(s', \mathbf{i}_{\mathbf{m}as'}, a'; \beta) - Q(s, \mathbf{i}_{\mathbf{m}}, a; \beta)$
      $\xi = \lambda\xi + \mathbf{y}(\mathbf{i}_{\mathbf{m}}) \otimes \mathbf{x}(s, a)$
      $\xi' = \lambda\xi + \mathbf{y}(\mathbf{i}_{\mathbf{m}as'}) \otimes \mathbf{x}(s', a')$
      $\beta \leftarrow \beta + \alpha\delta\xi - \alpha\xi'(\xi^\top\rho)$
      $\rho \leftarrow \rho + \omega(\delta J - \xi^\top\rho)\xi$
      $s \leftarrow s'$, $a \leftarrow a'$, $\tilde{t} \leftarrow \tilde{t} + 1$
    end while
  end while
  return $\arg\max_{a_t} Q(s_t, \mathbf{i}_{\mathbf{m}_t}, a_t; \beta)$
end procedure

Bayes-Adaptive Temporal Difference Search

Inputs:
  s_t, $\mathbf{i}_{\mathbf{m}_t}$
Initialize:
  $\beta \leftarrow 0$, $\rho \leftarrow 0$
procedure SEARCH(s_t, $\mathbf{i}_{\mathbf{m}_t}$)
  while time remaining **do** ▷ Start episode
    $s \leftarrow s_t$, $\mathbf{i}_{\mathbf{m}} \leftarrow \mathbf{i}_{\mathbf{m}_t}$, $\tilde{t} \leftarrow t$
    $a \leftarrow \hat{\pi}_{\epsilon\text{–greedy}}(s, \mathbf{i}_{\mathbf{m}}; Q)$
    $\xi \leftarrow 0$, $\xi' \leftarrow 0$
    while $\tilde{t} < T$ **do**
      $s' \sim \hat{\mathcal{T}}(s, \mathbf{i}_{\mathbf{m}}, a, \cdot, \mathbf{i}_{\mathbf{ma}\cdot})$
      $R \leftarrow \mathcal{R}(s, \mathbf{i}_{\mathbf{m}}, a)$
      $a' \leftarrow \hat{\pi}_{\epsilon\text{–greedy}}(s', \mathbf{i}_{\mathbf{m}as'}; Q)$
      $\delta \leftarrow R + Q(s', \mathbf{i}_{\mathbf{m}as'}, a'; \beta) - Q(s, \mathbf{i}_{\mathbf{m}}, a; \beta)$
      $\xi = \lambda\xi + \mathbf{y}(\mathbf{i}_{\mathbf{m}}) \otimes \mathbf{x}(s, a)$
      $\xi' = \lambda\xi + \mathbf{y}(\mathbf{i}_{\mathbf{m}as'}) \otimes \mathbf{x}(s', a')$
      $\beta \leftarrow \beta + \alpha\delta\xi - \alpha\xi'(\xi^\top\rho)$
      $\rho \leftarrow \rho + \omega(\delta J - \xi^\top\rho)\xi$
      $s \leftarrow s'$, $a \leftarrow a'$, $\tilde{t} \leftarrow \tilde{t} + 1$
    end while
  end while
  return $\arg\max_{a_t} Q(s_t, \mathbf{i}_{\mathbf{m}_t}, a_t; \beta)$
end procedure

Bayes-Adaptive Temporal Difference Search

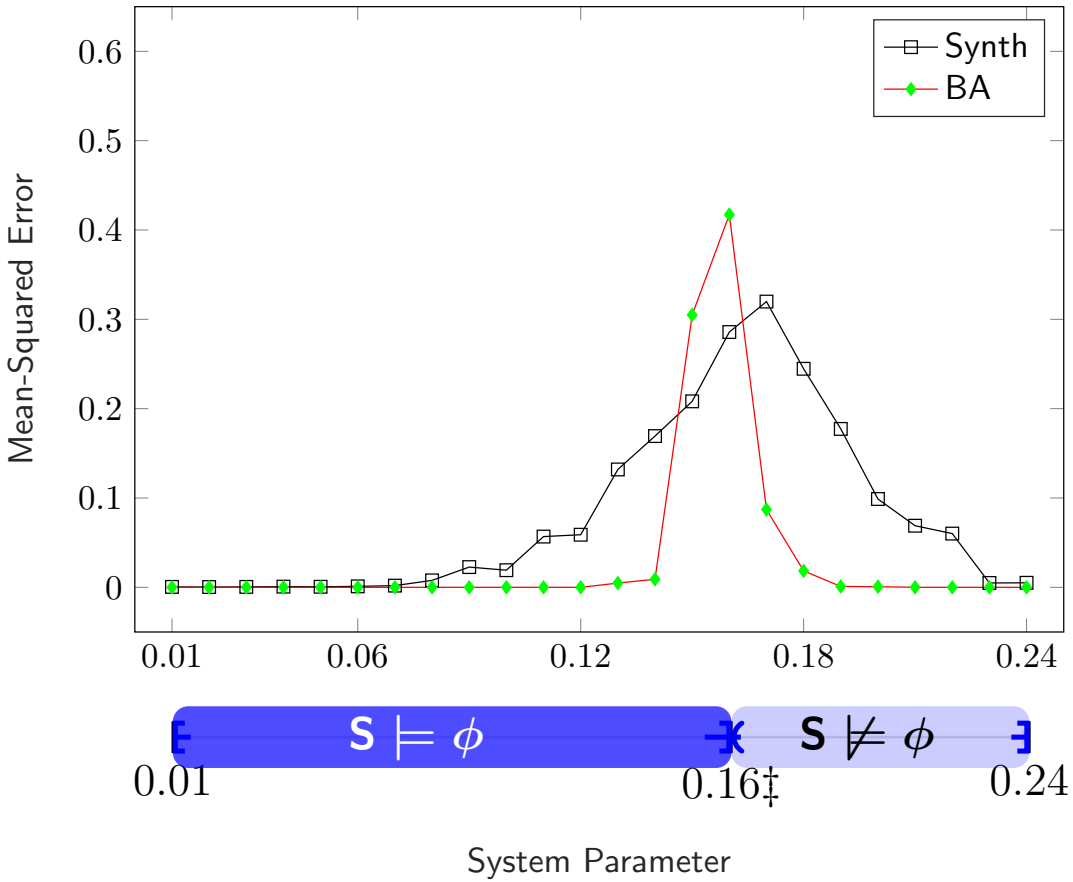
Inputs:
  s_t, $\mathrm{i}_{\mathbf{m}_t}$
Initialize:
  $\beta \leftarrow 0$, $\rho \leftarrow 0$
procedure SEARCH(s_t, $\mathrm{i}_{\mathbf{m}_t}$)
  while time remaining **do** ▷ Start episode
    $s \leftarrow s_t$, $\mathrm{i}_{\mathbf{m}} \leftarrow \mathrm{i}_{\mathbf{m}_t}$, $\tilde{t} \leftarrow t$
    $a \leftarrow \hat{\pi}_{\epsilon\text{–greedy}}(s, \mathrm{i}_{\mathbf{m}}; Q)$
    $\xi \leftarrow 0$, $\xi' \leftarrow 0$
    while $\tilde{t} < T$ **do**
      $s' \sim \hat{\mathcal{T}}(s, \mathrm{i}_{\mathbf{m}}, a, \cdot, \mathrm{i}_{\mathbf{ma}\cdot})$
      $R \leftarrow \mathcal{R}(s, \mathrm{i}_{\mathbf{m}}, a)$
      $a' \leftarrow \hat{\pi}_{\epsilon\text{–greedy}}(s', \mathrm{i}_{\mathbf{m}as'}; Q)$
      $\delta \leftarrow R + Q(s', \mathrm{i}_{\mathbf{m}as'}, a'; \beta) - Q(s, \mathrm{i}_{\mathbf{m}}, a; \beta)$
      $\xi = \lambda\xi + \mathbf{y}(\mathrm{i}_{\mathbf{m}}) \otimes \mathbf{x}(s, a)$
      $\xi' = \lambda\xi + \mathbf{y}(\mathrm{i}_{\mathbf{m}as'}) \otimes \mathbf{x}(s', a')$
      $\beta \leftarrow \beta + \alpha\delta\xi - \alpha\xi'(\xi^\top\rho)$
      $\rho \leftarrow \rho + \omega(\delta J - \xi^\top\rho)\xi$
      $s \leftarrow s'$, $a \leftarrow a'$, $\tilde{t} \leftarrow \tilde{t} + 1$
    end while
  end while
  return $\arg\max_{a_t} Q(s_t, \mathrm{i}_{\mathbf{m}_t}, a_t; \beta)$
end procedure

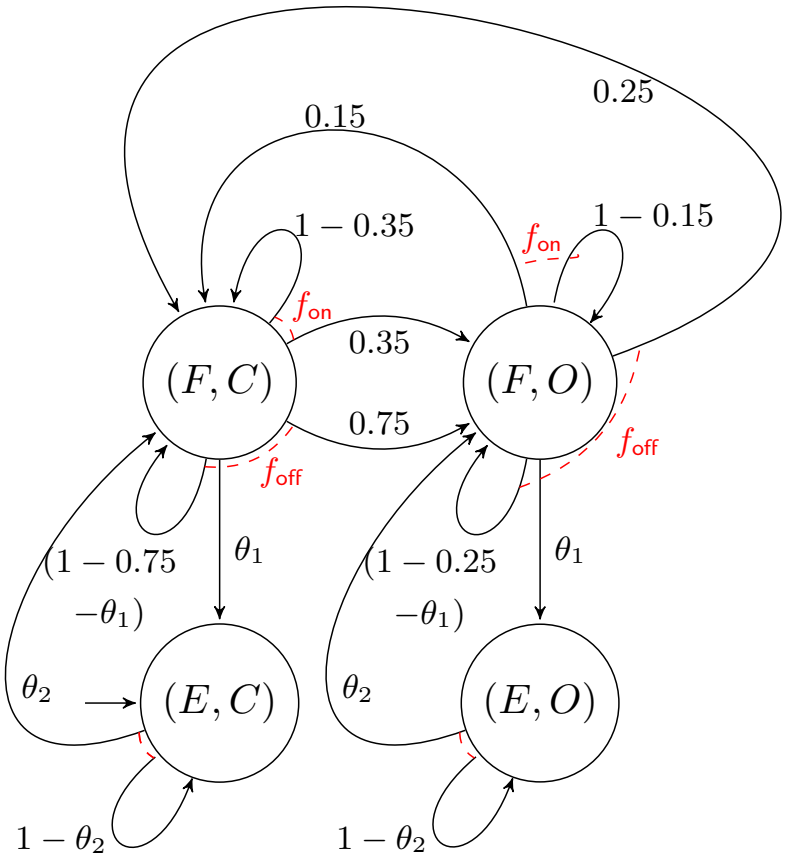
Outline

Evaluation - Case 2

$$\mathbf{P}_{\geq 0.35} \neg (\text{true } \mathcal{U}^{\leq 20} \ (\mathsf{E}, \mathsf{O}))$$

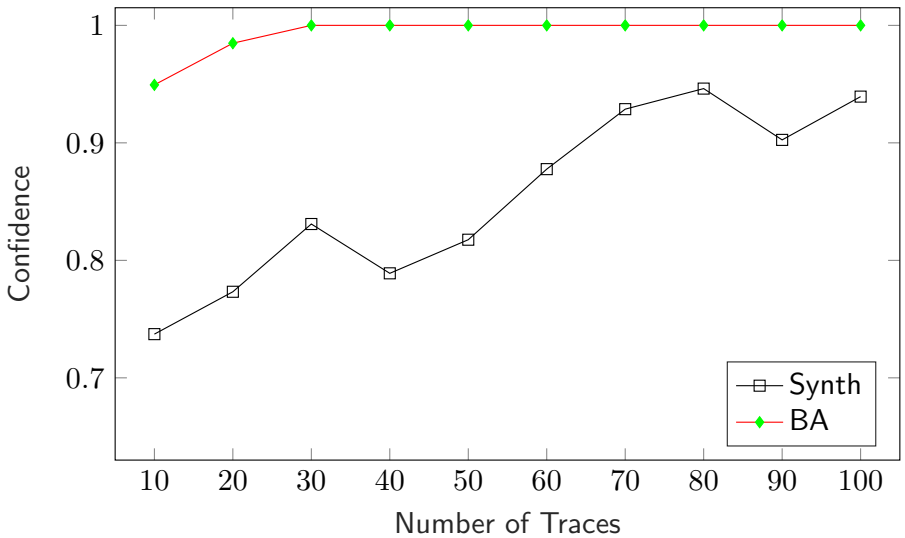
System Parameter $\theta_* = (\theta_1 = \theta_2)$

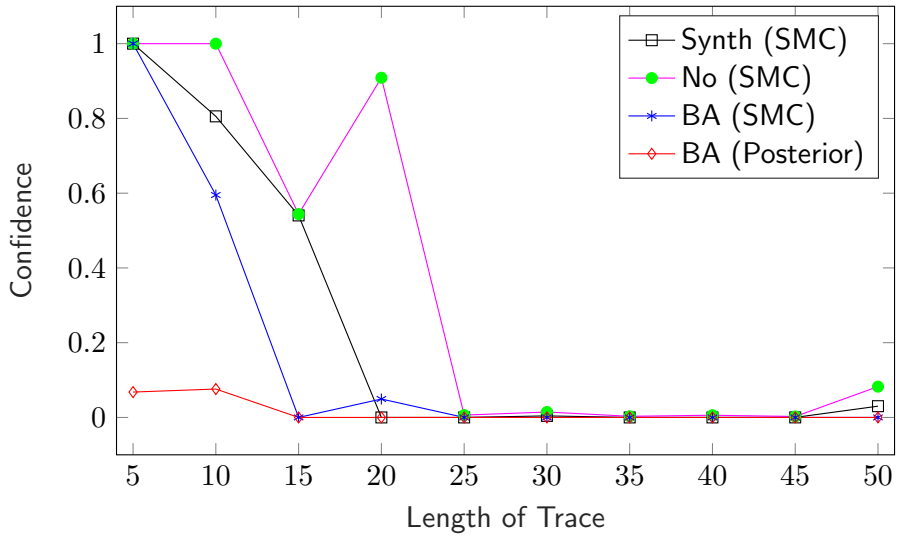




Evaluation - Case 2

$$\mathbf{P}_{\geq 0.35} \neg (\text{true } \mathcal{U}^{\leq 20} \text{ (E, O)})$$

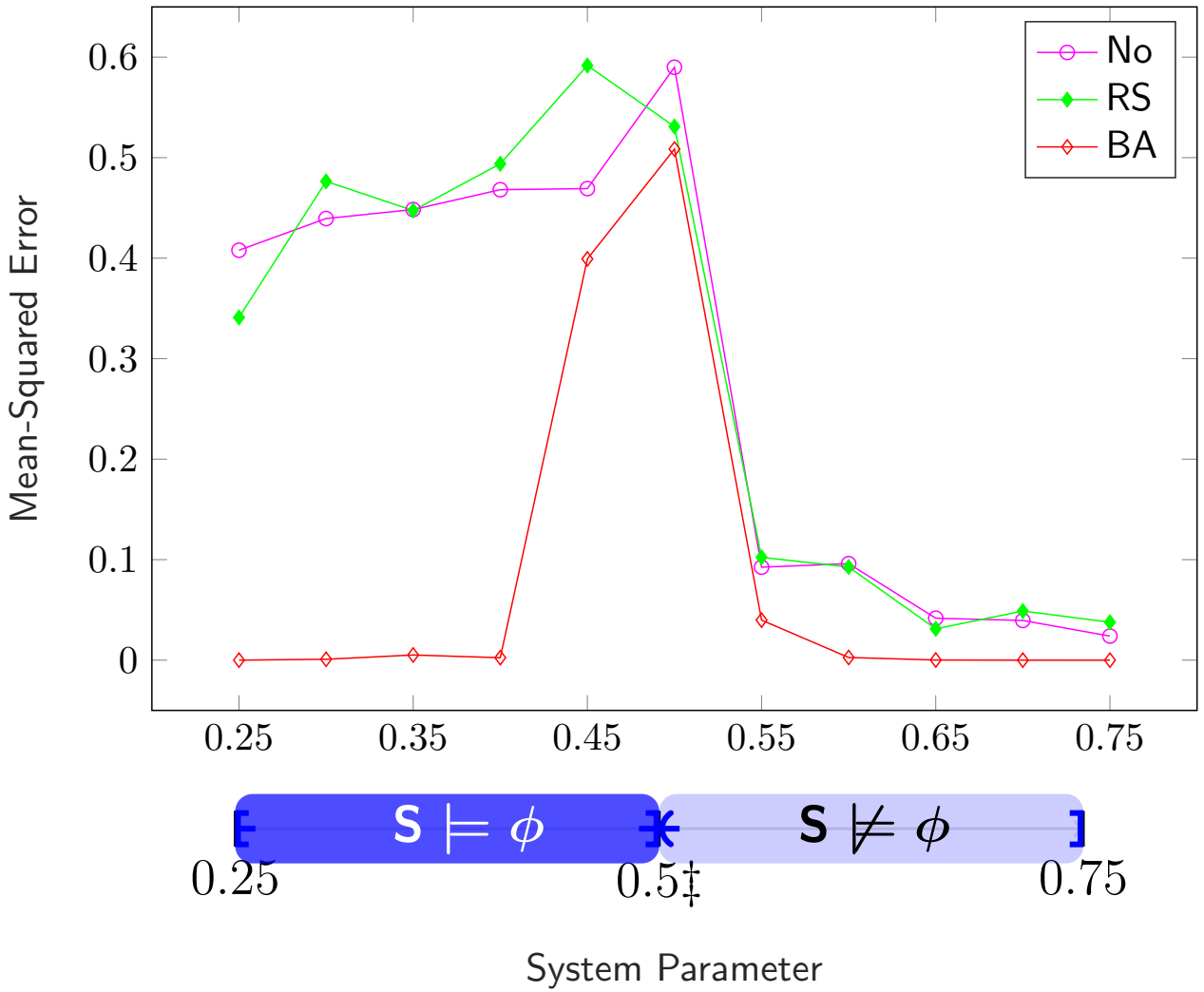




Evaluation - Case 3

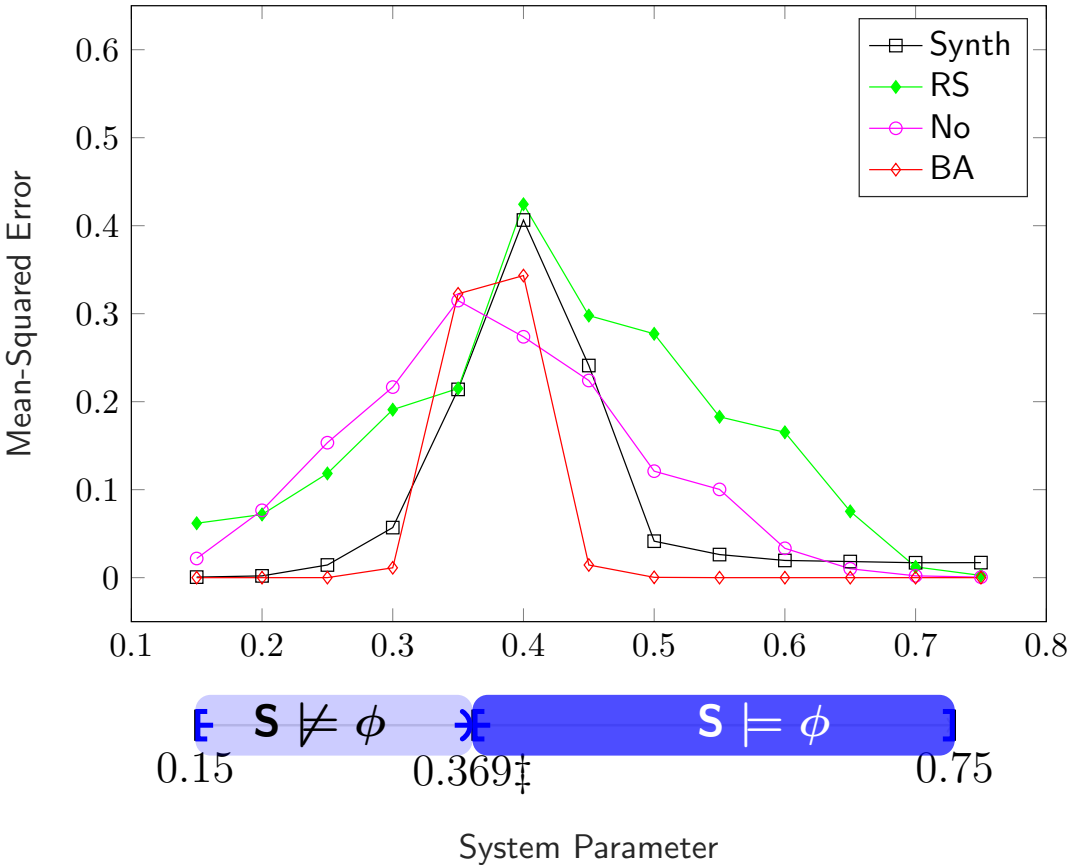
$\mathbf{P}_{\geq 0.25}(\text{true } \mathcal{U} \text{ (finished } \& \text{ allCoinsEqualToOne))}$

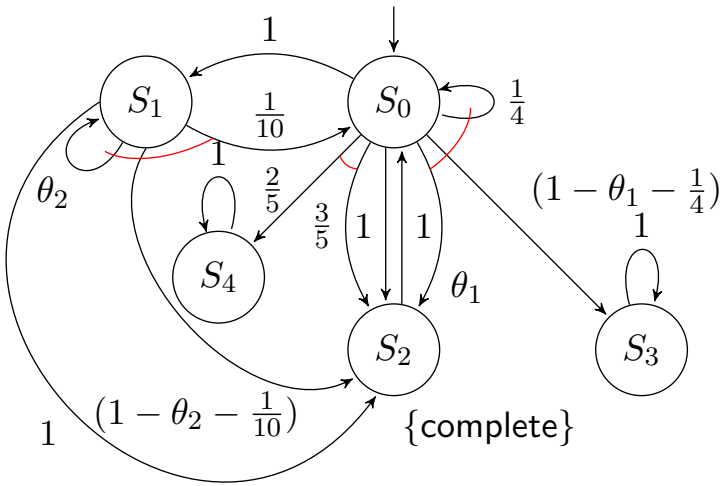
4112 states 7692 transitions



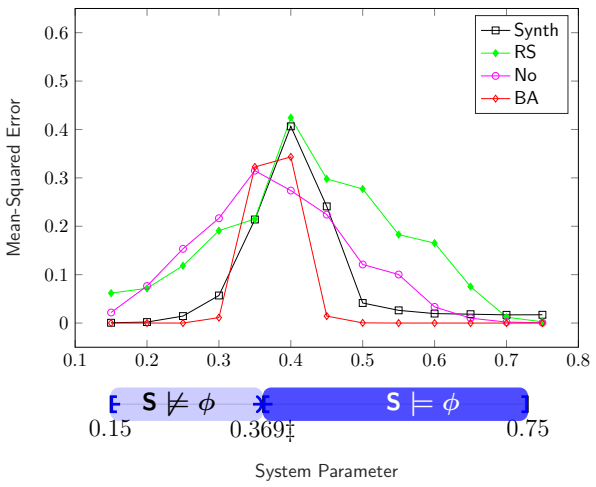
Evaluation - Case 1

$$\mathbf{P}_{\geq 0.5}(\text{true } \mathcal{U} \text{ complete})$$

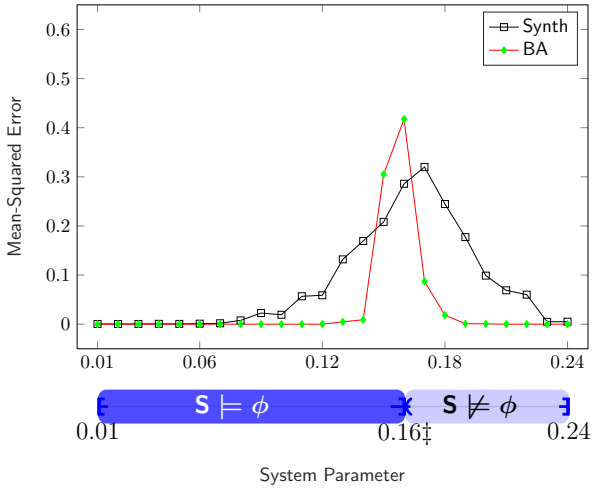




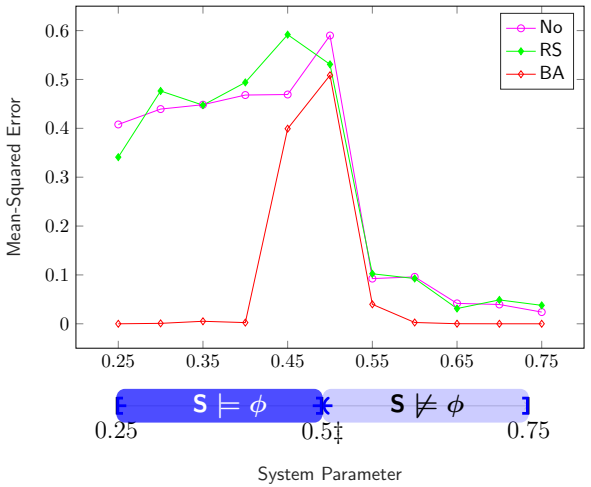
Evaluation - Summary



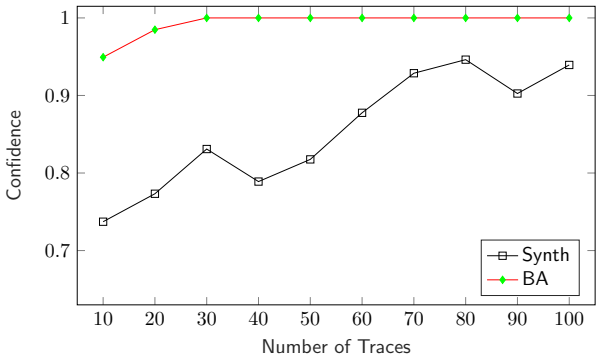
(a) MSE for Case 1 ($t10$,$l10$)



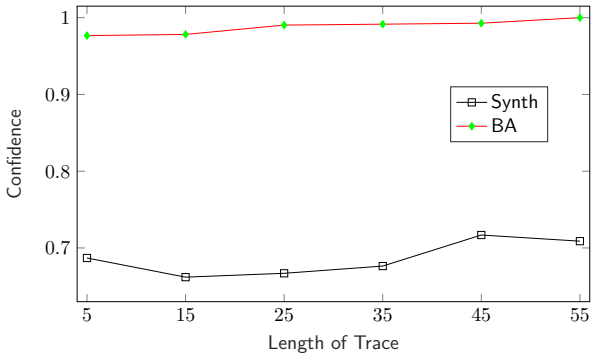
(b) MSE for Case 2 ($t10$,$l20$)



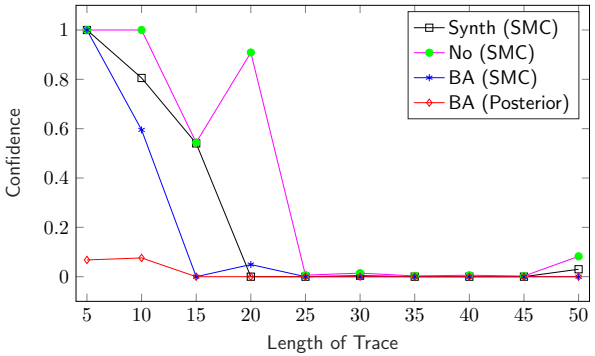
(c) MSE for Case 3 ($t10$,$l10$)



(d) Case 2 ($\theta_* = 0.13$,$l20$)



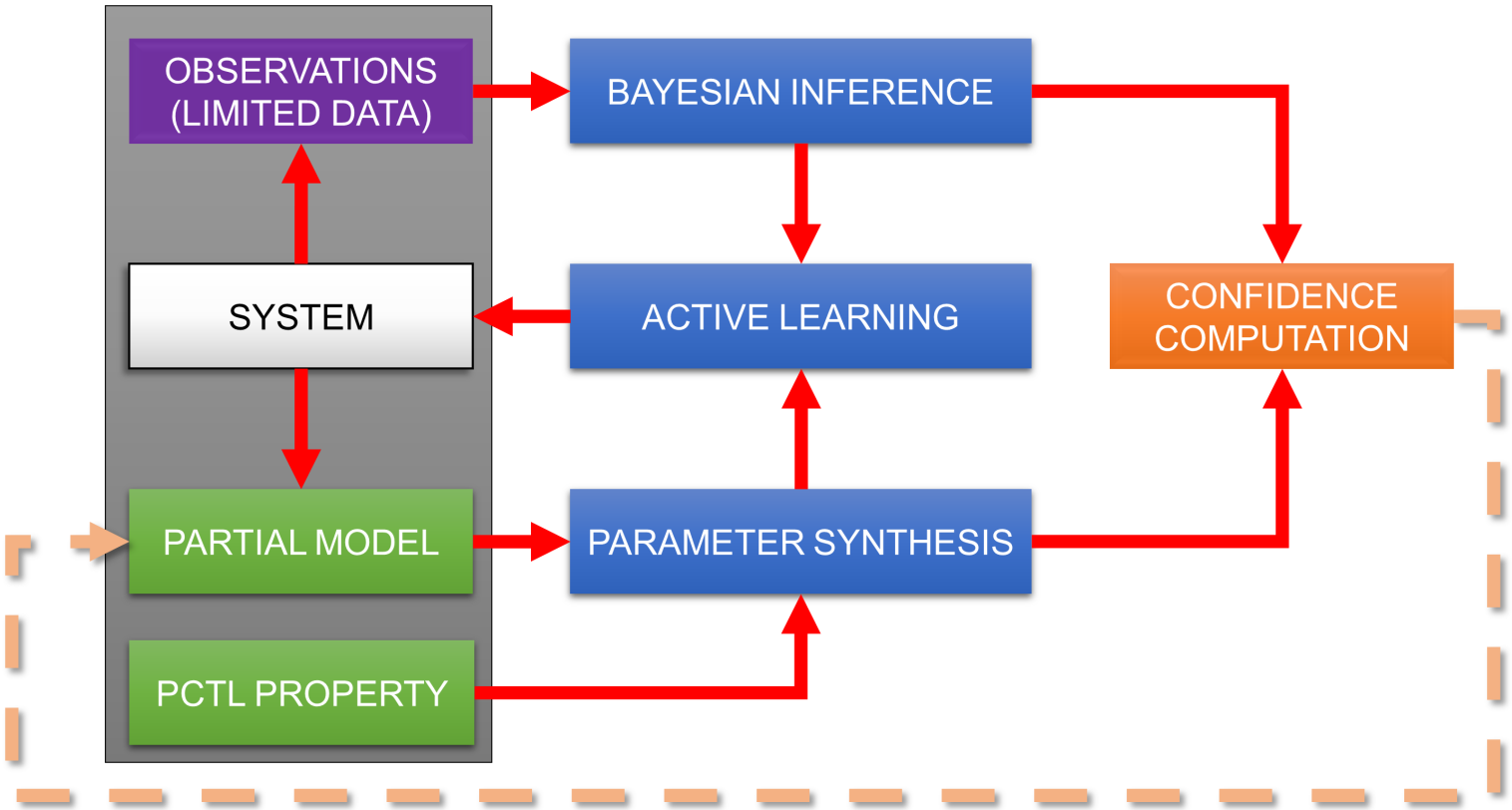
(e) Case 2 ($\theta_* = 0.14$,$t10$)



(f) Case 2 ($\theta_* = 0.24$,$t10$)

Takeaway Message

Verification of physical systems: Integrated learning and verification



Epilogue

Conclusions

- Our approach is able to gather more useful data
- Robust to the amount of data gathered
- Can work with much shorter horizons
- Can handle large MDP models
- Takes into account memory when calculating optimal strategies
- Actions are exclusively selected for learning tasks

Future Work

- Tune/repair model until a decisive confidence is achieved
- Actions can affect model classification as well

OXFORD CONTROL AND VERIFICATION GROUP (OXCAV)

Visit our group page: http://www.oxcav.org

PI: Prof. Alessandro Abate

Questions?

For more information:

Email: viraj.wijesuriya@cs.ox.ac.uk

References I


S. Haesaert, P. Van den Hof, and A. Abate, "Data-driven property verification of grey-box systems by bayesian experiment design," in *Proceedings of the American Control Conference (ACC2015), 1-3 July 2015, Chicago, USA*, (United States), pp. 1800–1805, Institute of Electrical and Electronics Engineers (IEEE), 7 2015.


E. Polgreen, V. B. Wijesuriya, S. Haesaert, and A. Abate, "Data-efficient bayesian verification of parametric markov chains," in *Quantitative Evaluation of Systems* (G. Agha and B. Van Houdt, eds.), (Cham), pp. 35–51, Springer International Publishing, 2016.


E. Polgreen, V. B. Wijesuriya, S. Haesaert, and A. Abate, "Automated experiment design for data-efficient verification of parametric markov decision processes," in *Quantitative Evaluation of Systems* (N. Bertrand and L. Bortolussi, eds.), (Cham), pp. 259–274, Springer International Publishing, 2017.