Formal verification of complex systems: model-based and data-driven methods

Alessandro Abate

Department of Computer Science, University of Oxford

Alan Turing Institute - Jan 12, 2018

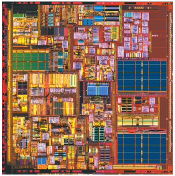
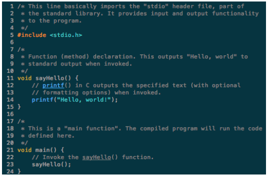
Automated formal verification: successes and frontiers

- automated, sound, formal

Automated formal verification: successes and frontiers

- automated, sound, formal
- industrial impact in verification of

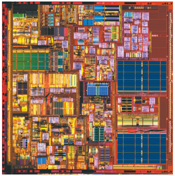
protocols, hardware circuits, and software

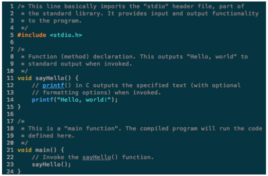




Automated formal verification: successes and frontiers

- automated, sound, formal
- industrial impact in verification of

protocols, hardware circuits, and software





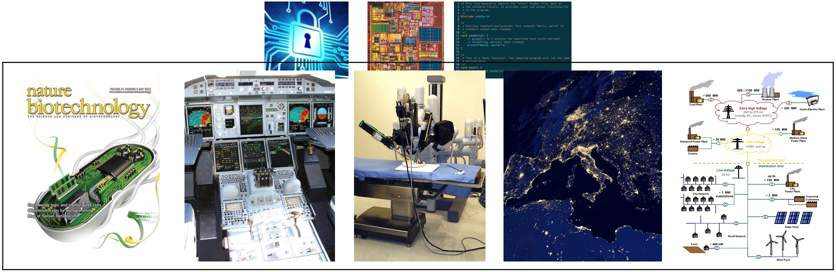
```c
/* This line basically imports the "stdio" header file, part of
 * the standard library. It provides input and output functionality
 * to the program.
 */
#include <stdio.h>

/*
 * Function (method) declaration. This outputs "Hello, world" to
 * standard output when invoked.
 */
void sayHello() {
    // printf() in C outputs the specified text (with optional
    // formatting options) when invoked.
    printf("Hello, world!");
}

/*
 * This is a "main function". The compiled program will run the code
 * defined here.
 */
void main() {
    // Invoke the sayHello() function.
    sayHello();
}
```

- asserts properties over given model of a system
- scalable and useful on "unsophisticated" models

Automated formal verification: pushing the envelope



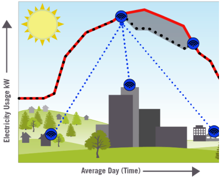
- verification of physical systems (cyber-physical systems)
  - dynamical models with uncertainty, noise (for CPS)
  - bridging the gap between data and models
  - principled integration of learning and verification

Building automation systems: an exemplar of CPS

- cyber-physical systems: integration of physical/analogue with cyber/digital
- building automation systems as a CPS exemplar

Building automation systems: an exemplar of CPS

- cyber-physical systems: integration of physical/analogue with cyber/digital
- building automation systems as a CPS exemplar



- *smart energy* initiatives at Oxford CS

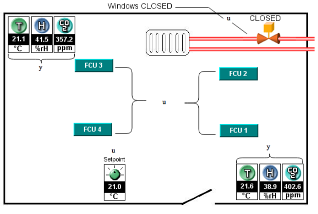
Building automation systems - a CPS exemplar

Building automation system setup in rooms 478/9 at Oxford CS

- advanced modelling for smart buildings
- application: certifiable energy management
  1. control of temperature, humidity, CO_2
  2. model-based predictive maintenance of devices
  3. fault-tolerant control
  4. demand-response over smart grids

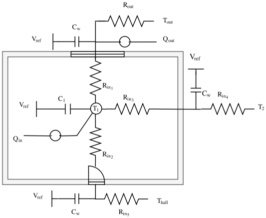
Building automation systems - a CPS exemplar



Building automation system setup in rooms 478/9 at Oxford CS

- advanced modelling for smart buildings
- application: certifiable energy management
  1. control of temperature, humidity, CO_2
  2. model-based predictive maintenance of devices
  3. fault-tolerant control
  4. demand-response over smart grids

Building automation systems - a CPS exemplar



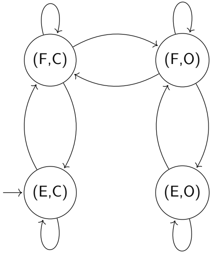
Building automation system setup in rooms 478/9 at Oxford CS

- advanced modelling for smart buildings
- application: certifiable energy management
  1. control of temperature, humidity, CO_2
  2. model-based predictive maintenance of devices
  3. fault-tolerant control
  4. demand-response over smart grids

Building automation systems - problem setup

- model CO_2 dynamics, under the effect of
  1. occupants: room full (F)/empty (E)
  2. window: open (O)/closed (C)
  3. air circulation: ON/OFF

$$x_{k+1} = x_k + \frac{\Delta}{V}\Big(-\mathbb{1}_{ON} m x_k + \mu_{\{O,C\}}(C_{out} - x_k)\Big) + \mathbb{1}_F C_{occ}$$

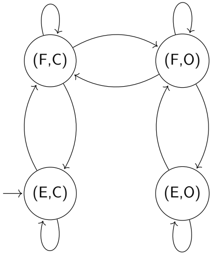


- x - zone CO_2 level
- Δ - sampling time
- V - zone volume
- m - air inflow (when ON)
- μ_O - air exchange with outside (when O)
- μ_C - air leakage with outside (when C)
- C_{out} - outside CO_2 level
- C_{occ} - CO_2 by occupants (when F)

Building automation systems - problem setup

- model CO_2 dynamics, under the effect of
  1. occupants: room full (F)/empty (E)
  2. window: open (O)/closed (C)
  3. air circulation: ON/OFF

$$x_{k+1} = x_k + \frac{\Delta}{V}\left(-\mathbb{1}_{ON} m x_k + \mu_{\{O,C\}}(C_{out} - x_k)\right) + \mathbb{1}_F C_{occ}$$

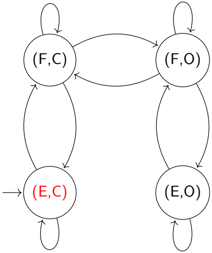


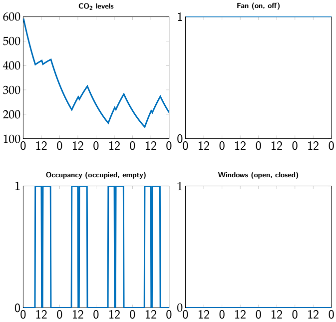
Parameter	Value
Δ	15 min
V	288 m^3
m	0.25 m^3/min
μ_O	0.1667 m^3/min
μ_C	0.01 m^3/min
C_{out}	375 ppm
C_{occ}	0.4 ppm/min

Building automation systems - problem setup

- model CO_2 dynamics, under the effect of
  1. occupants: room empty E
  2. window: closed C
  3. air circulation: ON

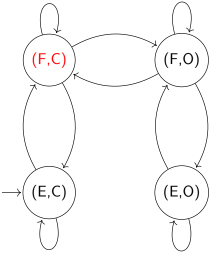
$$x_{k+1} = x_k + \frac{\Delta}{V}\left(-m x_k + \mu_C (C_{out} - x_k)\right) + 0 \cdot C_{occ}$$



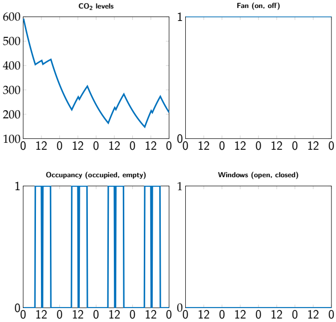


Building automation systems - problem setup

- model CO_2 dynamics, under the effect of
  1. occupants: room full F
  2. window: closed C
  3. air circulation: ON



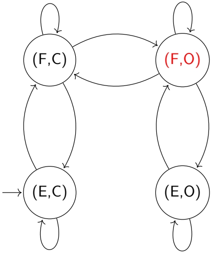
$$x_{k+1} = x_k + \frac{\Delta}{V}\left(-m x_k + \mu_C (C_{out} - x_k)\right) + C_{occ}$$

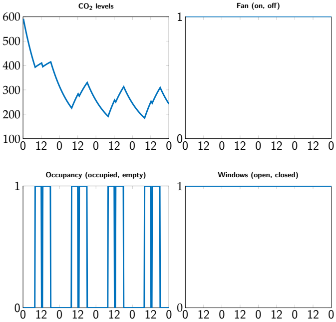


Building automation systems - problem setup

- model CO_2 dynamics, under the effect of
  1. occupants: room full F
  2. window: open O
  3. air circulation: ON

$$x_{k+1} = x_k + \frac{\Delta}{V}\left(-m x_k + \mu_O (C_{out} - x_k)\right) + C_{occ}$$

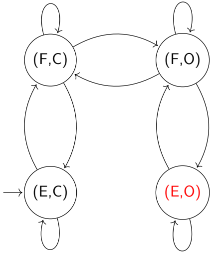


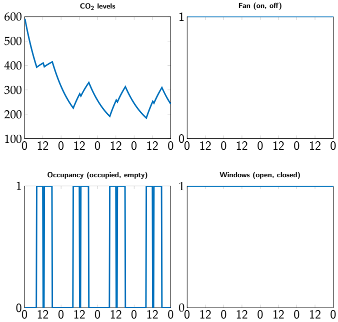


Building automation systems - problem setup

- model CO_2 dynamics, under the effect of
  1. occupants: room empty E
  2. window: closed C
  3. air circulation: ON

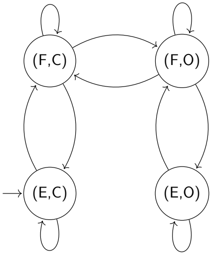
$$x_{k+1} = x_k + \frac{\Delta}{V}\left(-m x_k + \mu_O (C_{out} - x_k)\right)$$



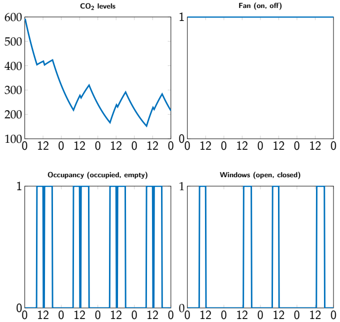


Building automation systems - problem setup

- model CO_2 dynamics, under the effect of
  1. occupants: room full (F)/empty (E)
  2. window: open (O)/closed (C)
  3. air circulation: ON

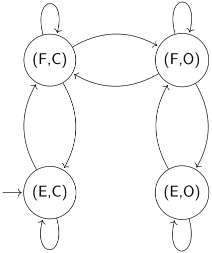


model with *hybrid* dynamics

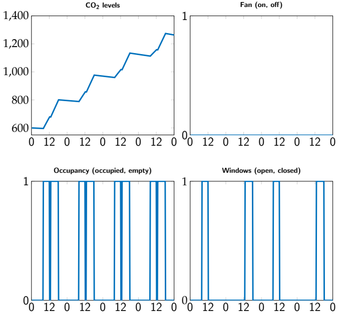


Building automation systems - problem setup

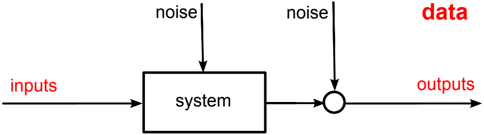
- model CO_2 dynamics, under the effect of
  1. occupants: room full (F)/empty (E)
  2. window: open (O)/closed (C)
  3. air circulation: OFF



model with *hybrid* dynamics

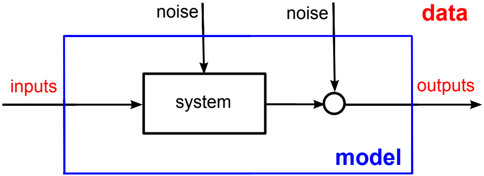


Learning and verification: state of art and objective



data-driven analysis

Learning and verification: state of art and objective

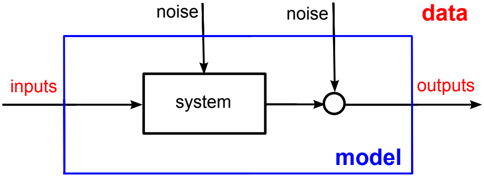


data-driven analysis

model learning (with data), and

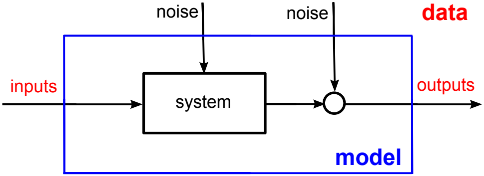
model-based verification

Learning and verification: state of art and objective



disconnect between data-driven learning and model-based verification

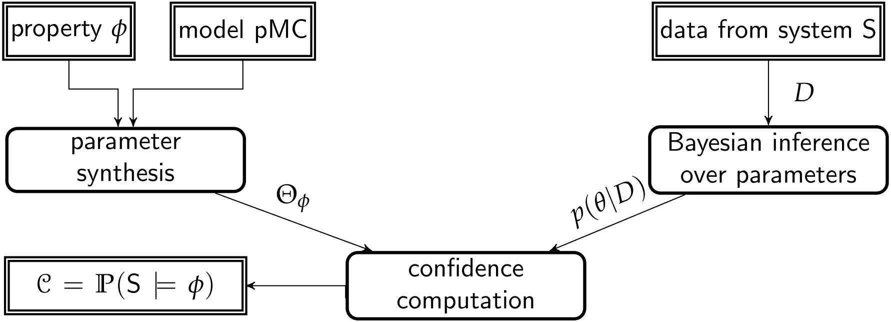
Learning and verification: state of art and objective



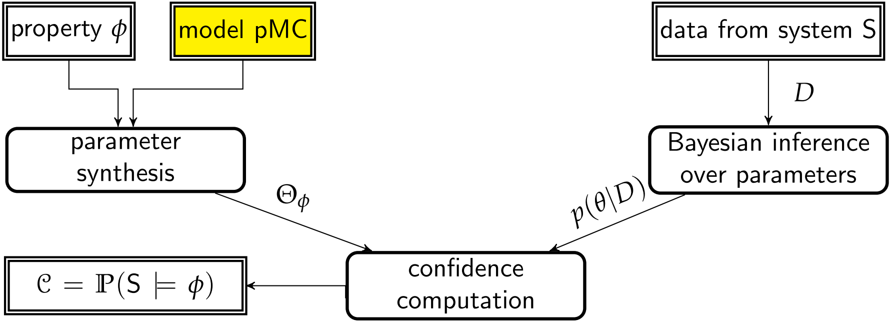
disconnect between data-driven learning and model-based verification

principled integration of learning and verification

Overview of method



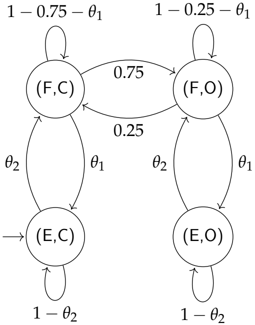
Parametric Markov chains



Parametric Markov chains

$$\mathcal{G} = (\Theta, S, \mathbb{T}_\theta, \rightarrow, \mathrm{AP}, L)$$

S – set of states
$\mathbb{T}_\theta$ – mapping $S \times S \rightarrow [0,1]$ expressed in terms of $\theta \in \Theta$
Θ – set of all possible valuations of θ, vector of parameters
$\rightarrow$ – starting states



$$\Theta = [0, 0.25] \times [0, 1]$$

Parametric Markov chains

$$\mathcal{G} = (\Theta, S, \mathbb{T}_\theta, \rightarrow, \mathrm{AP}, L)$$

S – set of states

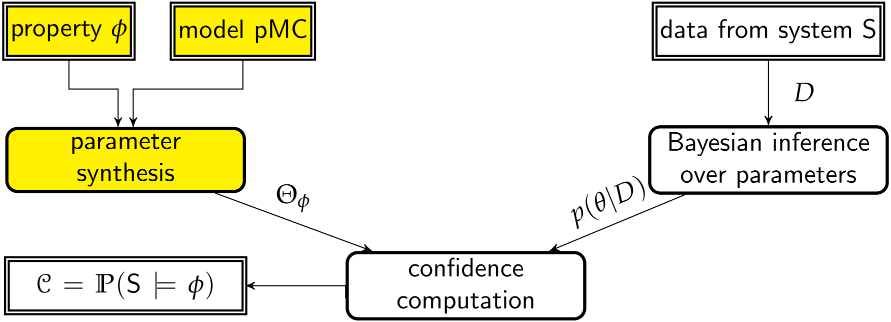
$\mathbb{T}_\theta$ – mapping $S \times S \rightarrow [0,1]$ expressed in terms of $\theta \in \Theta$

Θ – set of all possible valuations of θ, vector of parameters

$\rightarrow$ – starting states

L – labelling function, mapping states into 2^{AP}, AP alphabet

- denote by $M(\theta) \in \mathcal{G}$ a model parameterised by $\theta \in \Theta$



property ϕ
model pMC
data from system S
D
parameter synthesis
Bayesian inference over parameters
Θ_ϕ
$p(\theta|D)$
confidence computation
$\mathcal{C} = \mathbb{P}(\mathsf{S} \models \phi)$

- property ϕ specified in PCTL, e.g.

$$\phi = \mathbb{P}_{\geq 0.99}(\Box^{\leq 20} \text{ safe}), \qquad \phi = \mathbb{P}_{>0.5}(\text{safe U reach}), \qquad \text{safe}, \text{reach} \in \mathrm{AP}$$

- *probabilistic model checking* PCTL properties over Markov chains
  - `input:` Markov chain $(S, \mathbb{T})$, PCTL formula ϕ
  - `output:` $\mathsf{Sat}(\phi) = \{z \in S : z \models \phi\}$
- tools: PRISM, STORM, ...

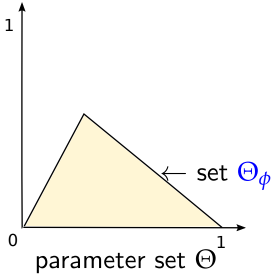
Parameter synthesis

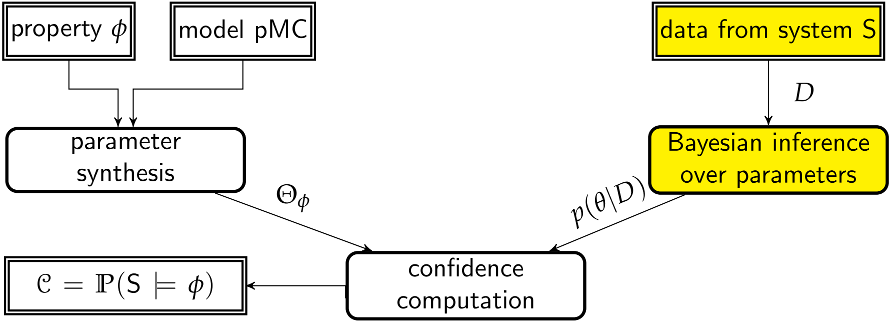
- property ϕ specified in PCTL, e.g.

$$\phi = \mathbb{P}_{\geq 0.99}(\Box^{\leq 20} \text{ safe}), \qquad \phi = \mathbb{P}_{>0.5}(\text{safe U reach}), \qquad \text{safe}, \text{reach} \in \mathrm{AP}$$

- classify models in Θ according to property of interest ϕ, that is
- synthesise parameters $\theta \in \Theta$ s.t. $M(\theta)$ satisfies ϕ:

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




property ϕ
model pMC
data from system S
D
parameter synthesis
Bayesian inference over parameters
Θ_ϕ
$p(\theta|D)$
confidence computation
$\mathcal{C} = \mathbb{P}(\mathsf{S} \models \phi)$

Bayesian inference

$$p(\theta_j \mid D) = \frac{\mathbb{P}(D \mid \theta_j) p(\theta_j)}{\mathbb{P}(D)}$$

$$= \frac{\prod_{s' \in S} \mathbb{T}_\theta(s_j, s')^{D_{s_j}^{s'}} \; p(\theta_j)}{\mathbb{P}(D_{s_j})}$$

- D – overall data gathered (traces)
  D_{s_j} – traces crossing state s_j, where $\theta_j = \theta_{s_j}$
- $p(\theta_j)$ – prior distribution
- $\prod_{s' \in S} \mathbb{T}_\theta(s_j, s')^{D_{s_j}^{s'}}$ – likelihood, multinomial distribution at state s_j

Bayesian inference

$$p(\theta_j \mid D) = \frac{\mathbb{P}(D \mid \theta_j) p(\theta_j)}{\mathbb{P}(D)}$$

$$= \frac{\prod_{s' \in S} \mathbb{T}_\theta(s_j, s')^{D_{s_j}^{s'}} \; p(\theta_j)}{\mathbb{P}(D_{s_j})}$$

- D – overall data gathered (traces)
  D_{s_j} – traces crossing state s_j, where $\theta_j = \theta_{s_j}$
- select as conjugate prior the Dirichlet distribution

$$p(\theta_j) = \mathrm{Dir}(\theta_j \mid \alpha) \propto \theta_j^{\alpha_1 - 1}(1 - \theta_j)^{\alpha_2 - 1}$$

for pair $(\theta_j, 1 - \theta_j)$, with $\alpha = (\alpha_1, \alpha_2)$ hyperparameters

Bayesian inference

$$p(\theta_j \mid D) = \frac{\mathbb{P}(D \mid \theta_j) p(\theta_j)}{\mathbb{P}(D)}$$

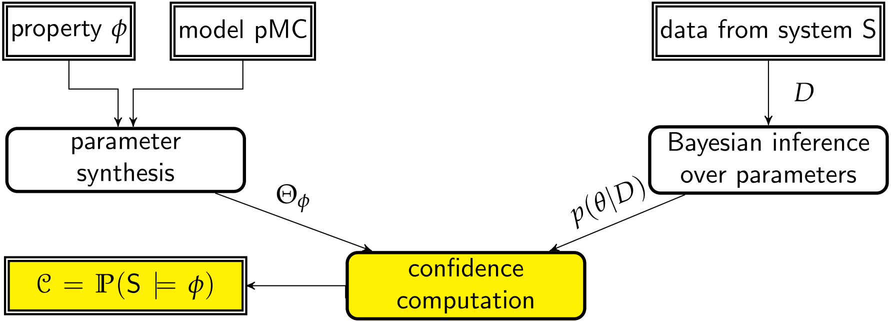
$$= \frac{\prod_{s' \in S} \mathbb{T}_\theta(s_j, s')^{D_{s_j}^{s'}} \; p(\theta_j)}{\mathbb{P}(D_{s_j})}$$

- D – overall data gathered (traces)
  D_{s_j} – traces crossing state s_j, where $\theta_j = \theta_{s_j}$
- under Dirichlet prior, posterior update is analytic

$$p(\theta_j \mid D) \propto \theta_j^{D_{s_j}^{s'_1}} (1-\theta_j)^{D_{s_j}^{s'_2}} \; \theta_j^{\alpha_1 - 1} (1-\theta_j)^{\alpha_2 - 1}$$

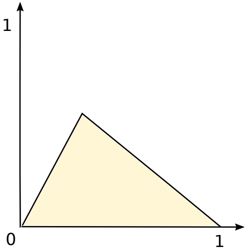
and obtained updating hyperparameters of Dirichlet distribution, as

$$p(\theta_j | D) = \mathrm{Dir}(\theta_j \mid D_{s_j} + \alpha)$$

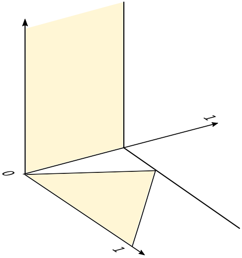


property ϕ
model pMC
data from system S
D
parameter synthesis
Bayesian inference over parameters
Θ_ϕ
$p(\theta|D)$
$\mathcal{C} = \mathbb{P}(\mathsf{S} \models \phi)$
confidence computation

Confidence computation



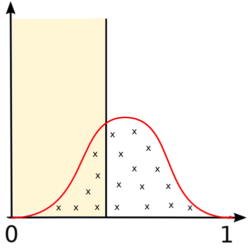
Confidence computation



Confidence computation

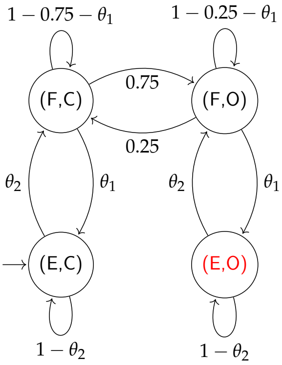
- compute confidence $\mathcal{C}$ on whether system S satisfies property ϕ as

$$\mathcal{C} = \mathbb{P}(\mathsf{S} \models \phi \mid D) = \int_{\Theta_\phi} p(\theta \mid D) d\theta$$



Case study: setup

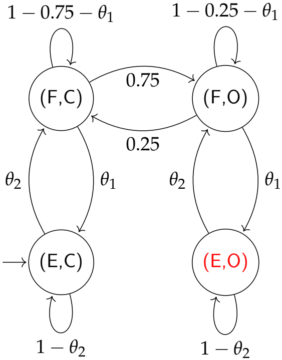
- goal: benchmark against statistical model checking (SMC)
- pMC model:

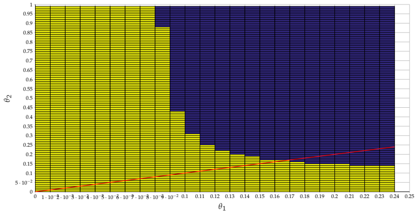


- specification: $\phi = \mathbb{P}_{>0.3}(\Box^{\leq 20} \neg(E,O))$

Case study: setup

- goal: benchmark against statistical model checking (SMC)
- pMC model:



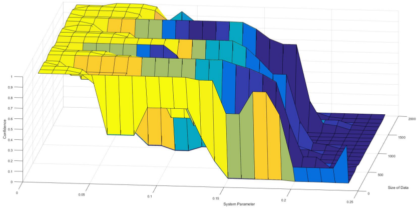


- specification: $\phi = \mathbb{P}_{>0.3}(\Box^{\leq 20} \neg(E,O))$
- for selected pMC and property, synthesis yields Θ_ϕ (yellow set)

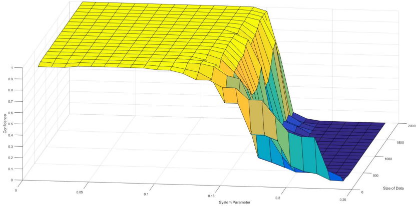
Case study: experiments

- data: state trajectories of different length

SMC



this work



- attains confidence closer to “true” value than SMC
- extracts information from data more efficiently
- is more robust with limited data

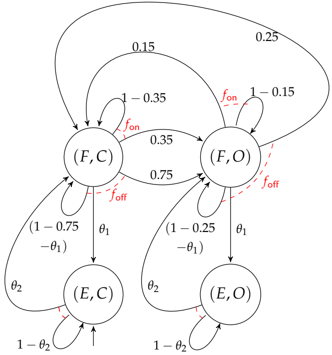
Parametric Markov decision processes

$$\mathcal{G} = (\Theta, S, A, \mathbb{T}_\theta, \rightarrow, \mathrm{AP}, L)$$

$\Theta, S, \rightarrow, L$ – as before

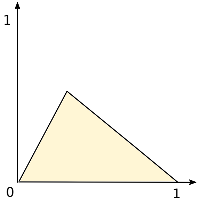
A – set of actions

$\mathbb{T}_\theta$ – mapping $S \times A \times S \rightarrow [0,1]$ expressed in terms of $\theta \in \Theta$



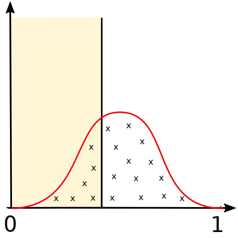
Dual role of actions in pMDP

- actions can be employed to shape set Θ_ϕ



shape set Θ_ϕ

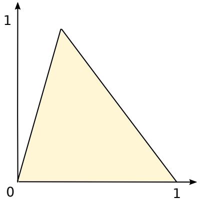
- actions can be chosen to affect confidence level $\mathcal{C}$



integral = confidence level

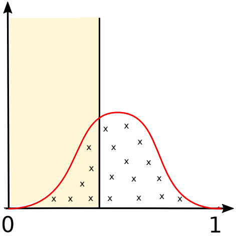
Dual role of actions in pMDP

- actions can be employed to shape set Θ_ϕ



shape set Θ_ϕ

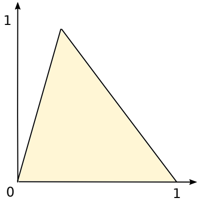
- actions can be chosen to affect confidence level $\mathcal{C}$



integral = confidence level

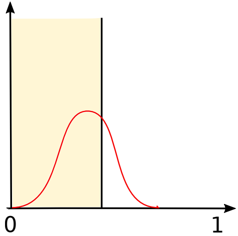
Dual role of actions in pMDP

- actions can be employed to shape set Θ_ϕ



shape set Θ_ϕ

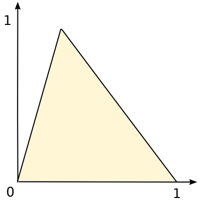
- actions can be chosen to affect confidence level $\mathcal{C}$



integral $\rightarrow$ confidence level

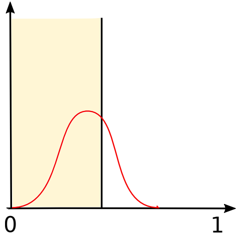
Dual role of actions in pMDP

- actions can be employed to shape set Θ_ϕ



shape set Θ_ϕ

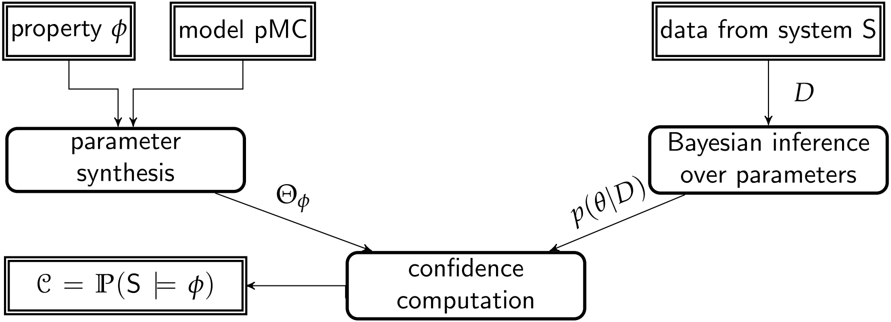
- actions can be chosen to affect confidence level $\mathcal{C}$



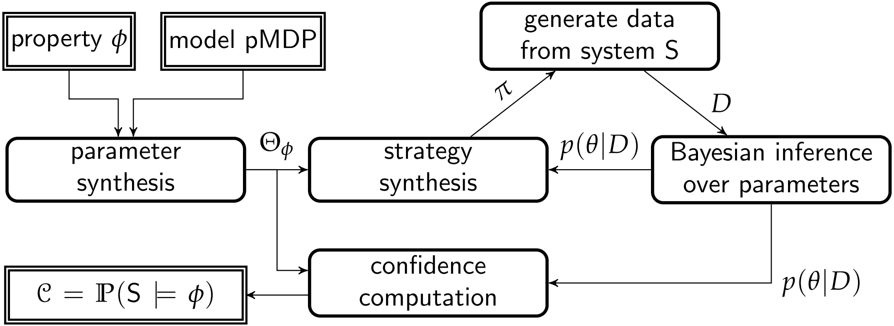
integral $\rightarrow$ confidence level

reminiscent of exploration/exploitation tradeoff in RL

Overview of method



Overview of method



Strategy synthesis for experiment design

- design experiments to affect confidence calculation

$$\max\left\{\mathbb{P}(\mathsf{S} \models \phi \mid D), \mathbb{P}(\mathsf{S} \not\models \phi \mid D)\right\}$$

Strategy synthesis for experiment design

- design experiments to affect confidence calculation

$$\max\{\mathbb{P}(\mathsf{S} \models \phi \mid D), \mathbb{P}(\mathsf{S} \not\models \phi \mid D)\}$$

- expected confidence gain at state-action (s, α) (and corresp. parameter)

$$\mathcal{C}_{s,\alpha} = \int_{\Theta_\phi} \prod_{\theta_i \in \theta} p(\theta_i \mid \mathbb{E}_{s,\alpha}(D_i)) d\theta$$

Strategy synthesis for experiment design

- design experiments to affect confidence calculation

$$\max \{\mathbb{P}(\mathsf{S} \models \phi \mid D), \mathbb{P}(\mathsf{S} \not\models \phi \mid D)\}$$

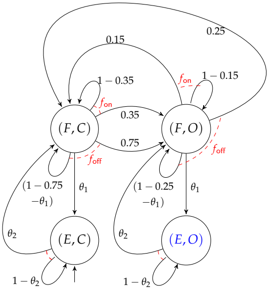
- expected confidence gain at state-action (s, α) (and corresp. parameter)

$$\mathcal{C}_{s,\alpha} = \int_{\Theta_\phi} \prod_{\theta_i \in \theta} p(\theta_i \mid \mathbb{E}_{s,\alpha}(D_i)) d\theta$$

- use $\mathcal{C}_{s,\alpha}$ as a reward for (s, α)
- synthesise optimal strategy π for experiment design

Case study: setup

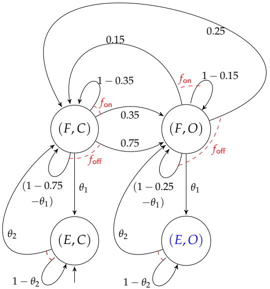
- goal: compare optimally synthesised policies vs. random/deterministic ones
- pMDP model:

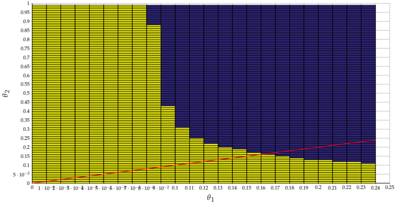


- specification: $\phi = \mathbb{P}_{>0.3}(\Box^{\leq 20} \neg(E, O))$

Case study: setup

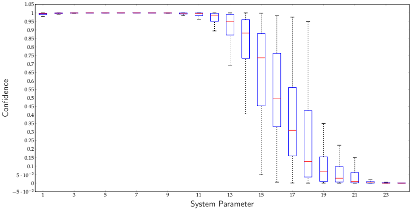
- goal: compare optimally synthesised policies vs. random/deterministic ones
- pMDP model:

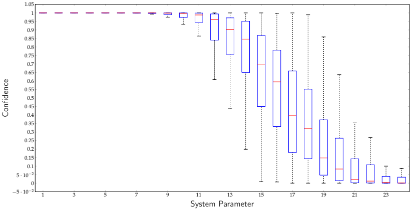




- specification: $\phi = \mathbb{P}_{>0.3}(\Box^{\leq 20} \neg(E, O))$
- for selected pMDP and given ϕ, Θ_ϕ is shown in yellow

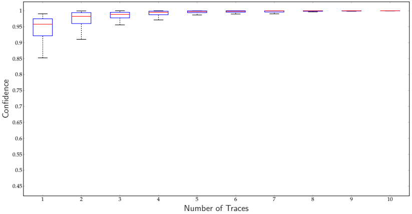
Case study: experiments

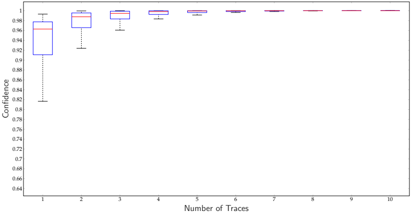




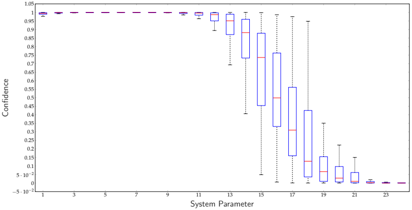
Synthesised strategy π

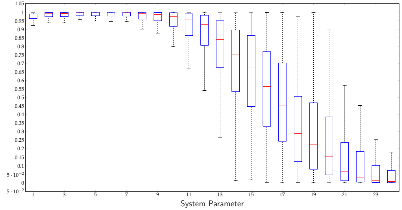
Fully random strategy





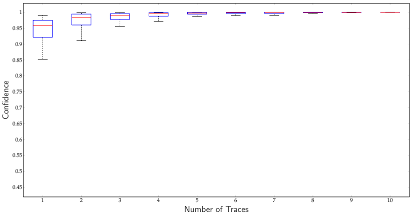
Case study: experiments

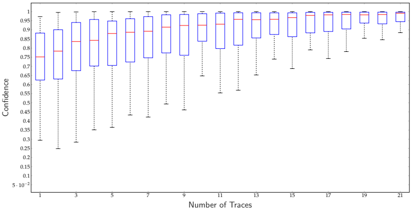




Synthesised strategy π

Deterministic strategy

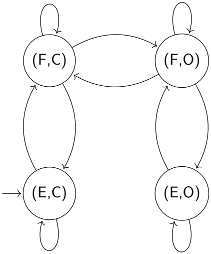




Extensions to other model classes

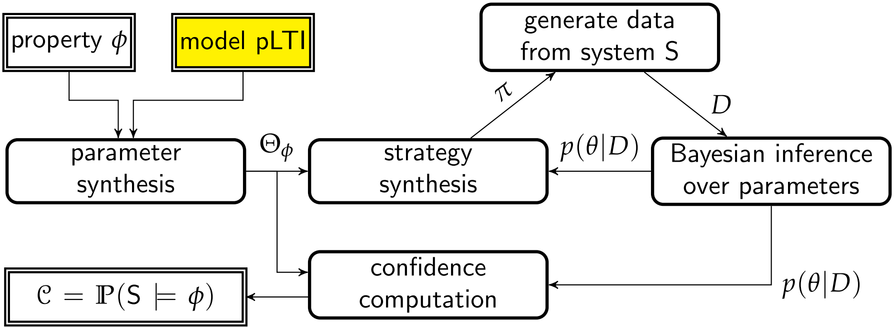
- model CO_2 dynamics, under the effect of
    1. occupants: room full (F)/empty (E)
    2. window: open (O)/closed (C)
    3. air circulation: ON/OFF

$$x_{k+1} = x_k + \frac{\Delta}{V}\left(-\mathbb{1}_{ON} m x_k + \mu_{\{O,C\}}(C_{out} - x_k)\right) + \mathbb{1}_F C_{occ}$$



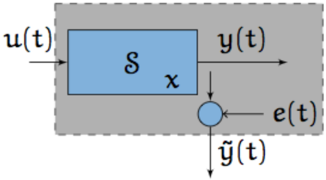
- x - zone CO_2 level
- Δ - sampling time
- V - zone volume
- m - air inflow (when ON)
- μ_O - air exchange with outside (when O)
- μ_C - air leakage with outside (when C)
- C_{out} - outside CO_2 level
- C_{occ} - CO_2 by occupants (when F)

Extensions to other model classes



Extensions to other model classes

- parametrised LTI model



$u(t)$ – input

$y(t)$ – system output

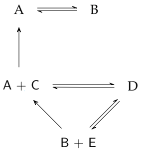
$\tilde{y}(t)$ – measured output

$e(t)$ – measurement noise, $e(t) \sim \mathcal{N}(0, \sigma_e^2)$

- model set $\mathcal{G} = \{M(\theta) \mid \theta \in \Theta\}$, where

$$M(\theta) : \quad \begin{cases} x(t+1) & = Ax(t) + Bu(t) \\ y(t) & = \theta^T x(t) \end{cases}$$

Applications of method



models for chemical reaction networks , with known stoichiometry, but with uncertain rates, expressed as pMDP

1. CRN can be excited by external input, pCT-MDP
2. limited data access (only to some states) to analyse known property
3. quantify confidence
4. synthesise optimal experiments
5. study actions tradeoff
6. if stoichiometry is not perfectly known, do network synthesis?

[red text: new theory needed]

Take away message

- integration of learning and verification
- verification and policy synthesis for Cyber-Physical Systems (CPS)
- application in Building Automation Systems (BAS)

Acknowledgments

My students: V. Wijesurija, N. Cauchi, E. Polgreen, A. Peruffo, K. Lesser, M. Zamani, S. Haesaert, I. Tkachev, D. Adzkiya, S. Soudjani and collaborators

Selected journal references

E. Polgreen, V. Wijesuriya, S. Haesaert and A. Abate, "Automated Experiment Design for Efficient Verification of Parametric Markov Decision Processes," QEST17, 2017.

E. Polgreen, V. Wijesuriya, S. Haesert and A. Abate, "Data-efficient Bayesian verification of parametric Markov chains," QEST16, LNCS 9826, pp. 35–51, 2016.

S. Haesaert, S.E.Z. Soudjani, and A. Abate, "Verification of general Markov decision processes by approximate similarity relations and policy refinement," SIAM Journal on Control and Optimisation, vol. 55, nr. 4, pp. 2333-2367, 2017.

I. Tkachev, A. Mereacre, J.-P. Katoen, and A. Abate, "Quantitative Model Checking of Controlled Discrete-Time Markov Processes," Information and Computation, vol. 253, nr. 1, pp. 1–35, 2017.

S. Haesaert, at al., P.M.J. V.d. Hof, and A. Abate, "Data-driven and Model-based Verification via Bayesian Identification and Reachability Analysis," Automatica, vol. 79, pp. 115–126, 2017.

S.E.Z. Soudjani and A. Abate, "Aggregation and Control of Populations of Thermostatically Controlled Loads by Formal Abstractions," IEEE Transactions on Control Systems Technology. vol. 23, nr. 3, pp. 975–990, 2015.

S.E.Z. Soudjani and A. Abate, "Quantitative Approximation of the Probability Distribution of a Markov Process by Formal Abstractions," Logical Methods in Computer Science, Vol. 11, nr. 3, Oct. 2015.

M. Zamani, P. Mohajerin Esfahani, R. Majumdar, A. Abate, and J. Lygeros, "Symbolic control of stochastic systems via approximately bisimilar finite abstractions," IEEE Transactions on Automatic Control, vol. 59 nr. 12, pp. 3135-3150, Dec. 2014.

I. Tkachev and A. Abate, "Characterization and computation of infinite horizon specifications over Markov processes," Theoretical Computer Science, vol. 515, pp. 1-18, 2014.

S. Soudjani and A. Abate, "Adaptive and Sequential Gridding for Abstraction and Verification of Stochastic Processes," SIAM Journal on Applied Dynamical Systems, vol. 12, nr. 2, pp. 921-956, 2013.

A. Abate, et al., "Approximate Model Checking of Stochastic Hybrid Systems," European Journal of Control, 16(6), 624-641, 2010.

A. Abate, et al., "Probabilistic Reachability and Safety Analysis of Controlled Discrete-Time Stochastic Hybrid Systems," Automatica, 44(11), 2724-2734, Nov. 2008.

Thank you for your attention

For more info: `aabate@cs.ox.ac.uk`