Parametric Statistical model checking of UAV flight plan[1]

Ran Bao[12] Christian Attiogbé[2] *Benoît Delahaye*[2]

[1]PIXIEL GROUP, Nantes, France

[2]Université de Nantes - LS2N, UMR 6004 - Nantes, France

SynCoP 2019

[1]Thanks to Didier Lime and Paulin Fournier

Motivation

UAVs flying above a crowd (Entertainment)

Motivation

UAVs flying above a crowd (Entertainment)

⇒ How to ensure that the flight is safe?

Contribution

We propose a model of the UAV system

- In the context of a flight plan
- Parametric: takes into account
  - Sensor precision and failure
  - Wind force
- Allows to predict the trajectory

We propose and use parametric statistical model checking techniques

- Computes an approximation of the probability of satisfying a property
  - as a parametric function
  - polynomial
  - with parametric confidence intervals

Outline

Outline

Markov Chains

Definition (Markov chain)

A *Markov chain* (MC, for short) is a tuple $\mathcal{M} = (S, s_0, P)$ where S is a denumerable set of states, $s_0 \in S$ is the initial state and $P : S \times S \to [0, 1]$ is the transition probability function.

- Finite run: $\rho = s_0 s_1 \dots s_n$ s.t. $P(s_i, s_{i+1}) > 0$
- $\Gamma(l)$: set of all runs of length l in $\mathcal{M}$
- Probability of a finite run: $\mathbb{P}_{\mathcal{M}}(\rho) = \Pi_{i=1}^{n} P(s_{i-1}, s_i)$

Properties

In the context of SMC, we only consider properties on bounded runs. Let $r : \Gamma(l) \to \mathbb{R}$ be a reward function.

Reachability $\mathbb{P}_{\mathcal{M}}(\Diamond^{\leq l} s)$. $\rho \models \Diamond^{\leq l} s$, if there exists $i \leq l$ such that $s_i = s$.

Safety $\mathbb{P}_{\mathcal{M}}(\Box^{=l} E)$. $\rho \models \Box^{=l} E$, if for all $i \leq l$, $s_i \in E$.

Expected reward $\mathbb{E}^{l}_{\mathcal{M}}(r)$. $\mathbb{E}^{l}_{\mathcal{M}}(r) = \sum_{\rho \in \Gamma(l)} \mathbb{P}_{\mathcal{M}}(\rho) r(\rho)$ is the expected value of r on the runs of length l.

Remark

For any property $\varphi \subseteq \Gamma(l)$, $\mathbb{P}_{\mathcal{M}}(\varphi) = \mathbb{E}^{l}_{\mathcal{M}}(\mathbb{1}_{\varphi})$
$\Rightarrow$ we focus on properties of the form $\mathbb{E}^{l}_{\mathcal{M}}(r)$.

Outline

Parametric Markov Chains (pMCs)

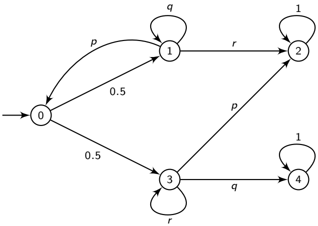
Definition (Parametric Markov chain)

A *Parametric Markov chain* is a tuple $\mathcal{M} = (S, s_0, P, \mathbb{X})$ such that S is a finite set of states, $s_0 \in S$ is the initial state, $\mathbb{X}$ is a finite set of parameters, and $P : S \times S \to Poly(\mathbb{X})$ is a parametric transition probability function.

If $v \in \mathbb{R}^{\mathbb{X}}$ is a valuation of the parameters,

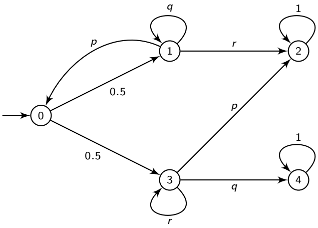
- P_v: transition probabilities under v : $P_v(s, s') = P(s, s')(v)$
- v is *valid* if (S, s_0, P_v) is a MC
- $\mathcal{M}^v = (S, s_0, P_v)$
- Runs and probabilities are similar to MC, but parametric

Example 1

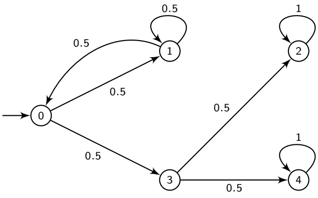


pMC $\mathcal{M}_1$

Example 1



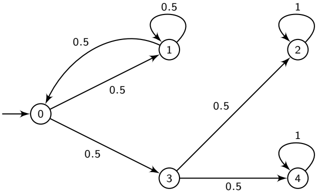
pMC $\mathcal{M}_1$



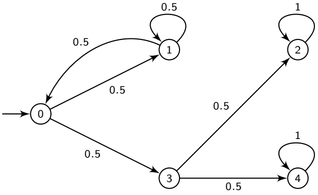
MC $\mathcal{M}_1^v$ for parameter valuation v such that $v(p) = v(q) = 0.5$ and $v(r) = 0$

Outline

Monte Carlo for MCs

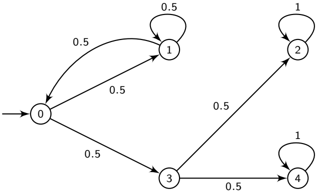


Monte Carlo for MCs



- Run n simulations ρ_i of length l

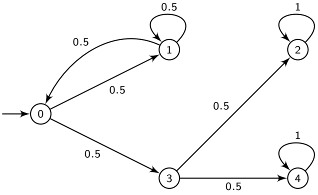
Monte Carlo for MCs



$\rho_1 = 0 \cdot 1 \cdot 1 \cdot 1 \cdot 1 \cdot 1$
$\rho_2 = 0 \cdot 1 \cdot 0 \cdot 3 \cdot 4 \cdot 4$
$\rho_3 = 0 \cdot 3 \cdot 2 \cdot 2 \cdot 2 \cdot 2$
$\rho_4 = 0 \cdot 1 \cdot 0 \cdot 1 \cdot 0 \cdot 3$
$\rho_5 = 0 \cdot 3 \cdot 4 \cdot 4 \cdot 4 \cdot 1$
$\rho_6 = 0 \cdot 3 \cdot 2 \cdot 2 \cdot 2 \cdot 2$
$\rho_7 = 0 \cdot 1 \cdot 0 \cdot 3 \cdot 2 \cdot 2$
$\rho_8 = 0 \cdot 1 \cdot 0 \cdot 3 \cdot 4 \cdot 4$

- Run n simulations ρ_i of length l

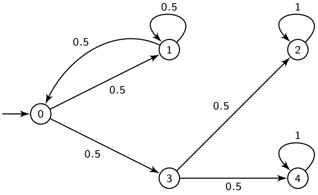
Monte Carlo for MCs



$$\begin{aligned}
\rho_1 &= 0 \cdot 1 \cdot 1 \cdot 1 \cdot 1 \cdot 1 \\
\rho_2 &= 0 \cdot 1 \cdot 0 \cdot 3 \cdot 4 \cdot 4 \\
\rho_3 &= 0 \cdot 3 \cdot 2 \cdot 2 \cdot 2 \cdot 2 \\
\rho_4 &= 0 \cdot 1 \cdot 0 \cdot 1 \cdot 0 \cdot 3 \\
\rho_5 &= 0 \cdot 3 \cdot 4 \cdot 4 \cdot 4 \cdot 1 \\
\rho_6 &= 0 \cdot 3 \cdot 2 \cdot 2 \cdot 2 \cdot 2 \\
\rho_7 &= 0 \cdot 1 \cdot 0 \cdot 3 \cdot 2 \cdot 2 \\
\rho_8 &= 0 \cdot 1 \cdot 0 \cdot 3 \cdot 4 \cdot 4
\end{aligned}$$

- Run n simulations ρ_i of length l
- $r(\rho_i) = 1$ if ρ_i reaches 4

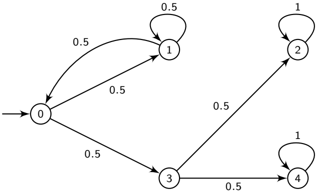
Monte Carlo for MCs



$$\begin{array}{ll}
\rho_1 = 0 \cdot 1 \cdot 1 \cdot 1 \cdot 1 \cdot 1 & r(\rho_1) = 0 \\
\rho_2 = 0 \cdot 1 \cdot 0 \cdot 3 \cdot 4 \cdot 4 & r(\rho_2) = 1 \\
\rho_3 = 0 \cdot 3 \cdot 2 \cdot 2 \cdot 2 \cdot 2 & r(\rho_3) = 0 \\
\rho_4 = 0 \cdot 1 \cdot 0 \cdot 1 \cdot 0 \cdot 3 & r(\rho_4) = 0 \\
\rho_5 = 0 \cdot 3 \cdot 4 \cdot 4 \cdot 4 \cdot 1 & r(\rho_5) = 1 \\
\rho_6 = 0 \cdot 3 \cdot 2 \cdot 2 \cdot 2 \cdot 2 & r(\rho_6) = 0 \\
\rho_7 = 0 \cdot 1 \cdot 0 \cdot 3 \cdot 2 \cdot 2 & r(\rho_7) = 0 \\
\rho_8 = 0 \cdot 1 \cdot 0 \cdot 3 \cdot 4 \cdot 4 & r(\rho_8) = 1
\end{array}$$

- Run n simulations ρ_i of length l
- $r(\rho_i) = 1$ if ρ_i reaches 4

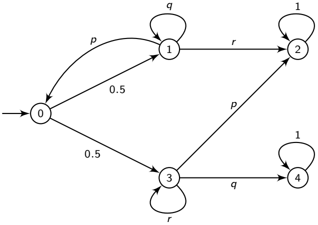
Monte Carlo for MCs



$$
\begin{array}{ll}
\rho_1 = 0\cdot1\cdot1\cdot1\cdot1\cdot1 & r(\rho_1) = 0 \\
\rho_2 = 0\cdot1\cdot0\cdot3\cdot4\cdot4 & r(\rho_2) = 1 \\
\rho_3 = 0\cdot3\cdot2\cdot2\cdot2\cdot2 & r(\rho_3) = 0 \\
\rho_4 = 0\cdot1\cdot0\cdot1\cdot0\cdot3 & r(\rho_4) = 0 \\
\rho_5 = 0\cdot3\cdot4\cdot4\cdot4\cdot1 & r(\rho_5) = 1 \\
\rho_6 = 0\cdot3\cdot2\cdot2\cdot2\cdot2 & r(\rho_6) = 0 \\
\rho_7 = 0\cdot1\cdot0\cdot3\cdot2\cdot2 & r(\rho_7) = 0 \\
\rho_8 = 0\cdot1\cdot0\cdot3\cdot4\cdot4 & r(\rho_8) = 1
\end{array}
$$

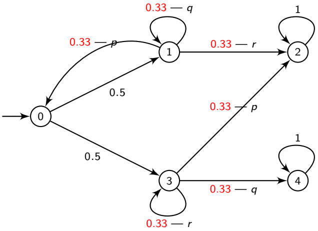
- Run n simulations ρ_i of length l
- $r(\rho_i) = 1$ if ρ_i reaches 4
- $\mathbb{E}^l_{\mathcal{M}}(r) \sim \frac{\sum r(\rho_i)}{n} \Rightarrow$ Here, $\mathbb{E}^5_{\mathcal{M}}(r) \sim 0.375$ (exact: 0.3125)

Intuition for pMCs



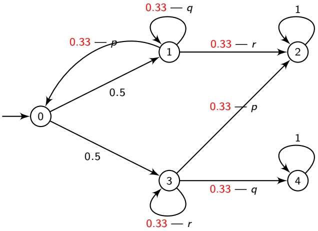
- How to run simulations?

Intuition for pMCs



- How to run simulations? Use a *normalization function* f (uniform?) $\rightarrow \mathcal{M}^f$

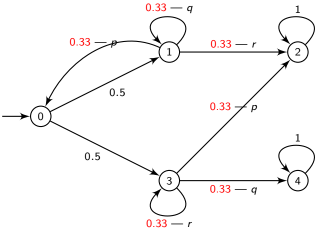
Intuition for pMCs



$\rho_1 = 0 \cdot 1 \cdot 1 \cdot 2 \cdot 2 \cdot 2$
$\rho_2 = 0 \cdot 1 \cdot 0 \cdot 3 \cdot 3 \cdot 4$
$\rho_3 = 0 \cdot 3 \cdot 2 \cdot 2 \cdot 2 \cdot 2$
$\rho_4 = 0 \cdot 1 \cdot 0 \cdot 1 \cdot 1 \cdot 0$
$\rho_5 = 0 \cdot 3 \cdot 4 \cdot 4 \cdot 4 \cdot 4$
$\rho_6 = 0 \cdot 3 \cdot 3 \cdot 3 \cdot 4 \cdot 4$
$\rho_7 = 0 \cdot 1 \cdot 0 \cdot 3 \cdot 2 \cdot 2$
$\rho_8 = 0 \cdot 1 \cdot 2 \cdot 2 \cdot 2 \cdot 2$

- How to run simulations? Use a *normalization function* f (uniform?) $\rightarrow \mathcal{M}^f$

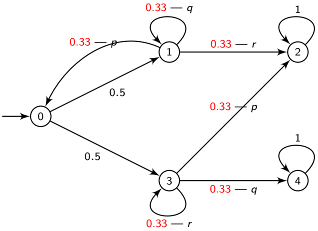
Intuition for pMCs



$\rho_1 = 0 \cdot 1 \cdot 1 \cdot 2 \cdot 2 \cdot 2$
$\rho_2 = 0 \cdot 1 \cdot 0 \cdot 3 \cdot 3 \cdot 4$
$\rho_3 = 0 \cdot 3 \cdot 2 \cdot 2 \cdot 2 \cdot 2$
$\rho_4 = 0 \cdot 1 \cdot 0 \cdot 1 \cdot 1 \cdot 0$
$\rho_5 = 0 \cdot 3 \cdot 4 \cdot 4 \cdot 4 \cdot 4$
$\rho_6 = 0 \cdot 3 \cdot 3 \cdot 3 \cdot 4 \cdot 4$
$\rho_7 = 0 \cdot 1 \cdot 0 \cdot 3 \cdot 2 \cdot 2$
$\rho_8 = 0 \cdot 1 \cdot 2 \cdot 2 \cdot 2 \cdot 2$

- How to run simulations? Use a *normalization function f* (uniform?) $\rightarrow \mathcal{M}^f$
- $R(\rho_i) = \mathbb{P}_{\mathcal{M}}(\rho)$ if ρ_i reaches 4, 0 otherwise

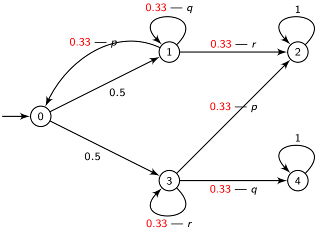
Intuition for pMCs



$$
\begin{array}{ll}
\rho_1 = 0 \cdot 1 \cdot 1 \cdot 2 \cdot 2 \cdot 2 & R(\rho_1) = 0 \\
\rho_2 = 0 \cdot 1 \cdot 0 \cdot 3 \cdot 3 \cdot 4 & R(\rho_2) = 0.25pqr \\
\rho_3 = 0 \cdot 3 \cdot 2 \cdot 2 \cdot 2 \cdot 2 & R(\rho_3) = 0 \\
\rho_4 = 0 \cdot 1 \cdot 0 \cdot 1 \cdot 1 \cdot 0 & R(\rho_4) = 0 \\
\rho_5 = 0 \cdot 3 \cdot 4 \cdot 4 \cdot 4 \cdot 4 & R(\rho_5) = 0.5q \\
\rho_6 = 0 \cdot 3 \cdot 3 \cdot 3 \cdot 4 \cdot 4 & R(\rho_6) = 0.5qr^2 \\
\rho_7 = 0 \cdot 1 \cdot 0 \cdot 3 \cdot 2 \cdot 2 & R(\rho_7) = 0 \\
\rho_8 = 0 \cdot 1 \cdot 2 \cdot 2 \cdot 2 \cdot 2 & R(\rho_8) = 0
\end{array}
$$

- How to run simulations? Use a *normalization function* f (uniform?) $\rightarrow \mathcal{M}^f$
- $R(\rho_i) = \mathbb{P}_{\mathcal{M}}(\rho)$ if ρ_i reaches 4, 0 otherwise

Intuition for pMCs



$$\begin{array}{ll}
\rho_1 = 0\cdot1\cdot1\cdot2\cdot2\cdot2 & R(\rho_1) = 0\\
\rho_2 = 0\cdot1\cdot0\cdot3\cdot3\cdot4 & R(\rho_2) = 0.25pqr\\
\rho_3 = 0\cdot3\cdot2\cdot2\cdot2\cdot2 & R(\rho_3) = 0\\
\rho_4 = 0\cdot1\cdot0\cdot1\cdot1\cdot0 & R(\rho_4) = 0\\
\rho_5 = 0\cdot3\cdot4\cdot4\cdot4\cdot4 & R(\rho_5) = 0.5q\\
\rho_6 = 0\cdot3\cdot3\cdot3\cdot4\cdot4 & R(\rho_6) = 0.5qr^2\\
\rho_7 = 0\cdot1\cdot0\cdot3\cdot2\cdot2 & R(\rho_7) = 0\\
\rho_8 = 0\cdot1\cdot2\cdot2\cdot2\cdot2 & R(\rho_8) = 0
\end{array}$$

- How to run simulations? Use a *normalization function f* (uniform?) $\rightarrow \mathcal{M}^f$
- $R(\rho_i) = \mathbb{P}_{\mathcal{M}}(\rho)$ if ρ_i reaches 4, 0 otherwise
- $\mathbb{E}^I_{\mathcal{M}}(R)$??

Results

Let $r'(\rho) = \frac{R(\rho)}{P_{\mathcal{M}^f}(\rho)} r(\rho)$. Due to the central limit theorem:

- $\mathbb{E}^l_{\mathcal{M}}(r')(v) = \mathbb{E}^l_{\mathcal{M}^v}(r)$ for all valid parameter valuation v
- Condition: Under v, $\mathcal{M}$ must conserve the same structure as under f

Results

Let $r'(\rho) = \frac{R(\rho)}{P_{\mathcal{M}^f}(\rho)} r(\rho)$. Due to the central limit theorem:

- $\mathbb{E}^l_{\mathcal{M}}(r')(v) = \mathbb{E}^l_{\mathcal{M}^v}(r)$ for all valid parameter valuation v
- Condition: Under v, $\mathcal{M}$ must conserve the same structure as under f

- Moreover, the confidence interval can be expressed as a polynomial function of the parameters. For n large enough and an error rate of 5%, its size is $3.92\widehat{\sigma}/\sqrt{n}$ where

$$\widehat{\sigma^2} = \frac{1}{(n-1)} \sum_{i=1}^{n} (r'(\rho_i))^2 - \frac{(\sum_{i=1}^{n} r'(\rho_i))^2}{n(n-1)}.$$

Results

Let $r'(\rho) = \frac{R(\rho)}{P_{\mathcal{M}^f}(\rho)} r(\rho)$. Due to the central limit theorem:

- $\mathbb{E}^I_{\mathcal{M}}(r')(v) = \mathbb{E}^I_{\mathcal{M}^v}(r)$ for all valid parameter valuation v
- Condition: Under v, $\mathcal{M}$ must conserve the same structure as under f

- Moreover, the confidence interval can be expressed as a polynomial function of the parameters. For n large enough and an error rate of 5%, its size is $3.92\widehat{\sigma}/\sqrt{n}$ where

$$\widehat{\sigma^2} = \frac{1}{(n-1)} \sum_{i=1}^{n} (r'(\rho_i))^2 - \frac{(\sum_{i=1}^{n} r'(\rho_i))^2}{n(n-1)}.$$

- Here, $\mathbb{E}^5_{\mathcal{M}}(r') \sim 3.48pqr + 0.38q + 3.47qr^2$.

Results

Let $r'(\rho) = \frac{R(\rho)}{P_{\mathcal{M}^f}(\rho)} r(\rho)$. Due to the central limit theorem:

- $\mathbb{E}^{l}_{\mathcal{M}}(r')(v) = \mathbb{E}^{l}_{\mathcal{M}^v}(r)$ for all valid parameter valuation v
- Condition: Under v, $\mathcal{M}$ must conserve the same structure as under f

- Moreover, the confidence interval can be expressed as a polynomial function of the parameters. For n large enough and an error rate of 5%, its size is $3.92\widehat{\sigma}/\sqrt{n}$ where

$$\widehat{\sigma^2} = \frac{1}{(n-1)} \sum_{i=1}^{n} (r'(\rho_i))^2 - \frac{(\sum_{i=1}^{n} r'(\rho_i))^2}{n(n-1)}.$$

- Here, $\mathbb{E}^{5}_{\mathcal{M}}(r') \sim 3.48pqr + 0.38q + 3.47qr^2$.
- For $v(p) = v(q) = 0.25, v(r) = 0.5$: $\mathbb{E}^{5}_{\mathcal{M}}(r')(v) \sim 0.42$ (exact: 0.261..)

Application to Example 1

For 1000 runs, we get

$$\mathbb{E}^5_{\mathcal{M}}(r') \sim 0.189 * p * q^2 + 0.405 * p * q * r + 0.252 * p * q + 0.729 * q * r^3 + 0.567 * q * r^2 + 0.54 * q * r + 0.492 * q$$

Application to Example 1

For 1000 runs, we get

$$\mathbb{E}^5_{\mathcal{M}}(r') \sim 0.189 * p * q^2 + 0.405 * p * q * r + 0.252 * p * q + 0.729 * q * r^3 + 0.567 * q * r^2 + 0.54 * q * r + 0.492 * q \sim 0.2801$$

Application to Example 1

For 1000 runs, we get

$$\mathbb{E}^5_{\mathcal{M}}(r') \sim 0.189 * p * q^2 + 0.405 * p * q * r + 0.252 * p * q + 0.729 * q * r^3 + 0.567 * q * r^2 + 0.54 * q * r + 0.492 * q \sim 0.2801$$

and the size of the CI is

$$(0.062 * (5.108 * p^2 * q^4 + 10.946 * p^2 * q^2 * r^2 + 2.27 * p^2 * q^2 + 59.108 * q^2 * r^6 + 15.324 * q^2 * r^4 + 4.865 * q^2 * r^2 + 1.478 * q^2 - 1.0 * (0.189 * p * q^2 + 0.405 * p * q * r + 0.252 * p * q + 0.729 * q * r^3 + 0.567 * q * r^2 + 0.54 * q * r + 0.492 * q)^2))^{\frac{1}{2}}$$

Application to Example 1

For 1000 runs, we get

$$\mathbb{E}^5_{\mathcal{M}}(r') \sim 0.189 * p * q^2 + 0.405 * p * q * r + 0.252 * p * q + 0.729 * q * r^3 + 0.567 * q * r^2 + 0.54 * q * r + 0.492 * q \sim 0.2801$$

and the size of the CI is

$$(0.062 * (5.108 * p^2 * q^4 + 10.946 * p^2 * q^2 * r^2 + 2.27 * p^2 * q^2 + 59.108 * q^2 * r^6 + 15.324 * q^2 * r^4 + 4.865 * q^2 * r^2 + 1.478 * q^2 - 1.0 * (0.189 * p * q^2 + 0.405 * p * q * r + 0.252 * p * q + 0.729 * q * r^3 + 0.567 * q * r^2 + 0.54 * q * r + 0.492 * q)^2))^{\frac{1}{2}} \sim 0.0296$$

Outline

Context: automated flight close to humans

Potential problems

- Physical Failure
  $\Rightarrow$ Risk of falling

Context: automated flight close to humans

Potential problems

- Physical Failure
  $\Rightarrow$ Risk of falling

Physical Failure

- Mitigated through redundancy
- Estimated/provided by constructor
- Easily analyzed
- Not always dramatic

Context: automated flight close to humans

Potential problems

- Physical Failure
  ⇒ Risk of falling
- Software Failure/Inaccuracy
  ⇒ Erroneous position

Physical Failure

- Mitigated through redundancy
- Estimated/provided by constructor
- Easily analyzed
- Not always dramatic

Context: automated flight close to humans

Potential problems

- Physical Failure
  $\Rightarrow$ Risk of falling
- Software Failure/Inaccuracy
  $\Rightarrow$ Erroneous position

Physical Failure

- Mitigated through redundancy
- Estimated/provided by constructor
- Easily analyzed
- Not always dramatic

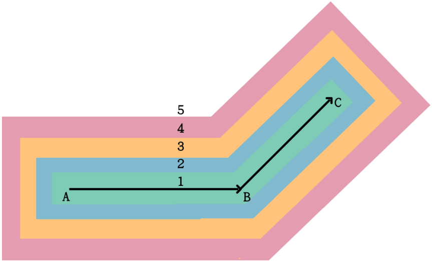
Guaranteeing human safety

- Estimating the position relies on several components
  - GPS
  - Gyroscope
  - Filters, etc.
- Precise position estimation ensures human safety

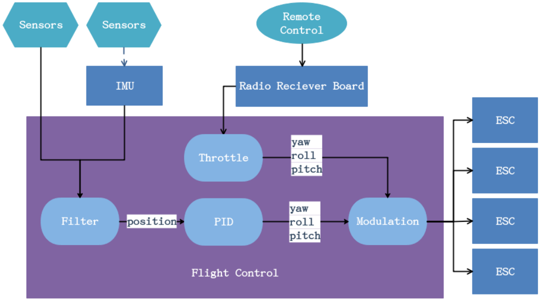
$\Rightarrow$ We will model and compute the probability that a UAV stays in a “safe zone”

Safety zones

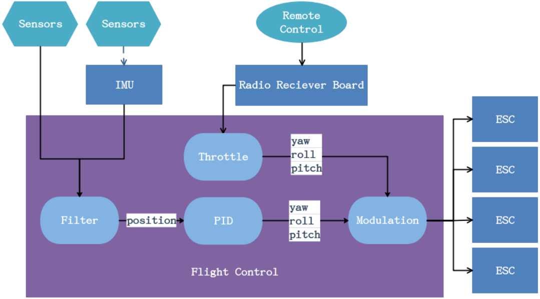
- Defined by distance from intended trajectory
- 5 zones
  - Inline with avionic certification (DO-178C)
- Fixed size



Main components

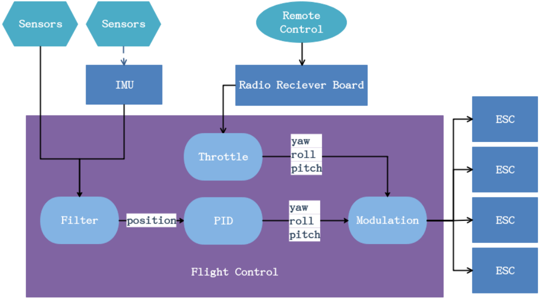


Main components



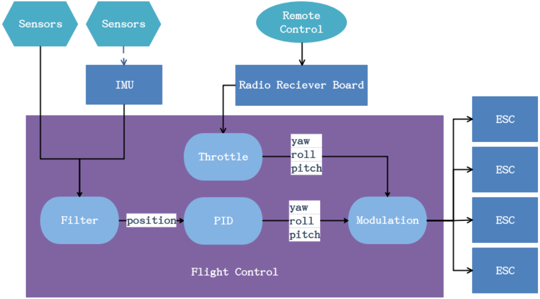
- Position estimation $=$ Sensors $+$ filter

Main components



- Position estimation = Sensors + filter
- Trajectory computation = PID

Main components



- Position estimation = Sensors + filter
- Trajectory computation = PID

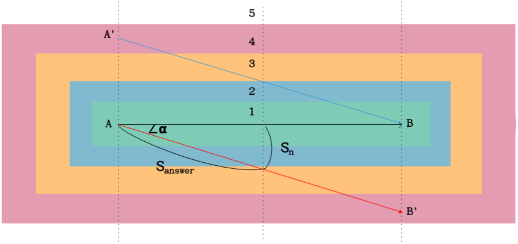
⇒ Parameters = Precision of (sensor + filter)

Position computation

Deviation is proportional to precision of position estimation

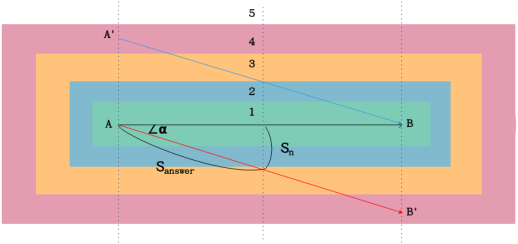
Position computation

Deviation is proportional to precision of position estimation



Position computation

Deviation is proportional to precision of position estimation



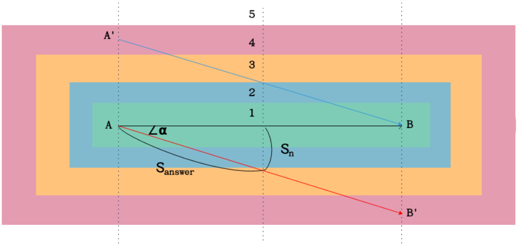
$$S_n = \frac{AA'}{T * f}$$

T = time to reach B

f = filter frequency

Position computation

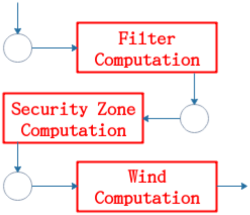
Deviation is proportional to precision of position estimation



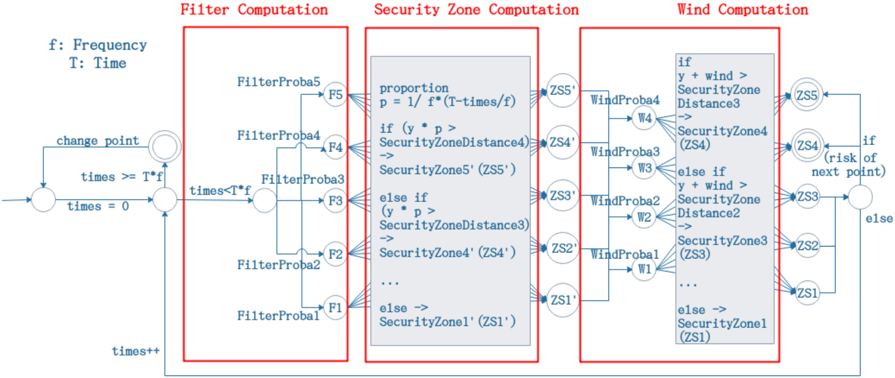
$$S_n = \frac{AA'}{T * f}$$

T = time to reach B

f = filter frequency



Resulting model



Outline

Prototype Implementation

MCpMC[a]

parametric Statistical Model Checking

- Written in Python
- input: prism model or python class
- output: parametric probability function / confidence interval

[a] Available at `https://github.com/Astlo/IMCpMC`

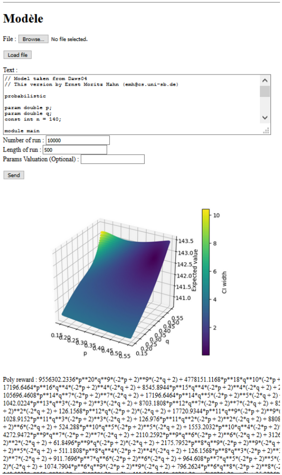
Prototype Implementation

MCpMC[a]

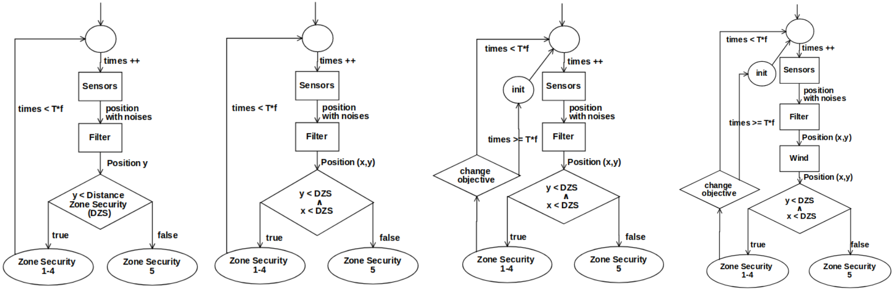
parametric Statistical Model Checking

- Written in Python
- input: prism model or python class
- output: parametric probability function / confidence interval
- graphical web interface (ongoing work)

[a] Available at https://github.com/Astlo/IMCpMC



Drone Model(s)



(a) Precision on y | (b) Precision on x, y | (c) Complex flight plan | (d) Wind disturbance

- 4 models of increasing complexity
- Position = real-valued variables
- Prism (a) and Python (b,c,d) models
- Parameters = precision of Position(x,y) + wind force

Results

	Model	10k		20k		50k	
		V1	**V2**	**V1**	**V2**	**V1**	**V2**
Running time	(c)	28s		51-54s		142-143s	
Senario 1	(c)	4.99%	5.09%	4.74%	5.10%	4.91%	4.98%
Senario 2	(c)	10.38%	10.04%	9.82%	10.05%	9.95%	9.81%
Running time	(d)(np)	28s		53-54s		149-155s	
Senario 1	(d)(np)	5.54%	5.19%	5.63%	5.72%	5.45%	5.51%
Senario 2	(d)(np)	10.7%	11.4%	11.0%	10.9%	10.9%	11.1%
Running time	(d)(p)	185-190s		311-314s		612-621s	
Senario 1	(d)(p)	5.18%	4.01%	3.54%	7.32%	6.96%	6.17%
Senario 2	(d)(p)	10.8%	9.20%	11.4%	8.54%	10.4%	12.2%

Precision parameters PF0/1/2/3/4: 0-2m / 2-4m / 4-6m / 6-8m / 8-10m
Zone4: 8m from trajectory. Zone5: 50m from trajectory

Scenario 1

- PF0/1/2/3/4 = 0.15/0.3/0.4/0.1/0.05

Scenario 2

- PF0/1/2/3/4 = 0.1/0.25/0.35/0.2/0.1

Outline

Improvements

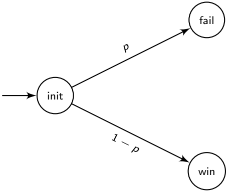
pSMC

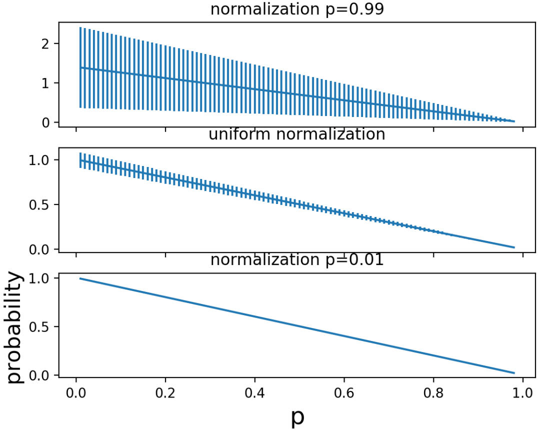
- Choice of normalization function
- Structure of the pMC
  If the structure is different under valuation and normalization, the estimated number of runs may change

UAV model

- Filter frequency as a parameter
- Complete flight plan

Impact of normalization function





Impact of the choice of the normalization function f on the size of confidence intervals.

- **Summary**:
  - Parametric Monte Carlo procedure for pMC
  - Polynomial parametric confidence interval;
  - Prototype implementation
  - Formal model of UAV flight plan
  - Parametric safety analysis

- **Summary**:
  - Parametric Monte Carlo procedure for pMC
  - Polynomial parametric confidence interval;
  - Prototype implementation

  - Formal model of UAV flight plan
  - Parametric safety analysis

- **Future work**:
  - Test and implement improvements
  - Comparison/Integration(?) to existing tools
  - UAV model improvements

- **Summary**:
  - Parametric Monte Carlo procedure for pMC
  - Polynomial parametric confidence interval;
  - Prototype implementation

  - Formal model of UAV flight plan
  - Parametric safety analysis

- **Future work**:
  - Test and implement improvements
  - Comparison/Integration(?) to existing tools
  - UAV model improvements

Thank you for your Attention