Numerical method to compute the law of exit times for piecewise deterministic Markov processes

Adrien Brandejsky, Benoîte de Saporta, François Dufour
University of Bordeaux and Inria
Outline

Motivation

Piecewise deterministic Markov processes

Exit time problem

Numerical results

Conclusion
Industrial problem from Astrium Space Transportation

Evaluate the **service time** of a material subject to **corrosion**

- support structure for other equipments
- small size: one point of measure
- long service life → monitor the **thickness** loss due to corrosion
Usage profile

Material subject to *corrosion*

**Usage profile**

Storage in 3 successive different environments for *random* times

1. workshop
2. submarine in operation
3. submarine in dry dock

Strong *safety* requirements

⇓

Compute the service time
Degradation process

- Deterministic succession of environments: $1 \rightarrow 2 \rightarrow 3 \rightarrow 1 \cdots$
Degradation process

- Deterministic succession of environments: $1 \rightarrow 2 \rightarrow 3 \rightarrow 1 \cdots$
- **Random** time in environment $i$ with distribution $\text{Exp}(\lambda_i)$
Degradation process

- Deterministic succession of environments: \(1 \rightarrow 2 \rightarrow 3 \rightarrow 1 \cdots\)
- Random time in environment \(i\) with distribution \(\text{Exp}(\lambda_i)\)
- Equation of thickness loss in environment \(i\)

\[
d_t = \rho_i \left( t - \eta_i + \eta_i \exp\left( -\frac{t}{2\eta_i} \right) \right)
\]

- \(\rho_i\) random corrosion rate in environment \(i\) with uniform distribution
- \(\eta_i\) deterministic time in environment \(i\).
Degradation process

- Deterministic succession of environments: $1 \rightarrow 2 \rightarrow 3 \rightarrow 1 \cdots$
- Random time in environment $i$ with distribution $\text{Exp}(\lambda_i)$
- Equation of thickness loss in environment $i$
  \[
  d_t = \rho_i \left( t - \eta_i + \eta_i \exp \left( -\frac{t}{2\eta_i} \right) \right)
  \]
- $\rho_i$ random corrosion rate in environment $i$ with uniform distribution
- $\eta_i$ deterministic time in environment $i$.

Inefficient material if $d_t \geq 0.2mm$
Simulated trajectories
Simulated trajectories
Aim of the collaboration with Astrium

Aims for Astrium
- compute the law of the service time
- assert that the structure will be safe for a given time with a given probability

Strategy
- model the corrosion problem with a PDMP
- give a general method to compute approximate exit time laws for PDMPs based on a special discretization of the process
Outline

Motivation

Piecewise deterministic Markov processes
  Definition
  Iterative construction

Exit time problem

Numerical results

Conclusion
Piecewise deterministic Markov processes

Davis (80’s)
General class of non-diffusion dynamic stochastic hybrid models: deterministic motion punctuated by random jumps.

Applications
Engineering systems, operations research, management science, economics, internet traffic, neurosciences, biology, dependability and safety, ...
Dynamics

Hybrid process $X_t = (m_t, y_t)$

- discrete mode $m_t \in \{1, 2, \ldots, p\}$
- Euclidean state variable $y_t \in \mathbb{R}^n$

Local characteristics for each mode $m$

- $E_m$ open subset of $\mathbb{R}^d$, $\partial E_m$ its boundary and $\overline{E}_m$ its closure
- Flow $\phi_m: \mathbb{R}^d \times \mathbb{R} \rightarrow \mathbb{R}^d$ deterministic motion between jumps, one-parameter group of homeomorphisms
- Intensity $\lambda_m: \overline{E}_m \rightarrow \mathbb{R}_+$ intensity of random jumps
- Markov kernel $Q_m$ on $(\overline{E}_m, \mathcal{B}(\overline{E}_m))$ selects post-jump location
Two types of jumps

- $t^*(m, y)$ deterministic exit time starting from $(m, y)$
  
  $$t^*(m, y) = \inf\{ t > 0 : \phi_m(y, t) \in \partial E_m \}$$

- law of the first jump time $T_1$ starting from $(m, y)$
  
  $$\mathbb{P}_{(m,y)}( T_1 > t) = \begin{cases} 
  e^{-\int_0^t \lambda_m(\phi_m(y, s))\, ds} & \text{if } t < t^*(m, y) \\
  0 & \text{if } t \geq t^*(m, y)
  \end{cases}$$

Remark

$T_1$ has a density on $[0, t^*(m, y)]$ but has an atom at $t^*(m, y)$

$$\mathbb{P}_{(m,y)}( T_1 = t^*(m, y)) > 0$$
Iterative construction

Starting point

\[ X_0 = Z_0 = (m, y) \]
Iterative construction

$X_t$ follows the deterministic flow until the first jump time $T_1 = S_1$

$$X_t = (m, \phi_m(y, t)), \quad t < T_1$$
Iterative construction

Post-jump location $Z_1 = (M_1, Y_1)$ selected by the Markov kernel

$Q_m(\phi_m(y, T_1), \cdot)$
Iterative construction

$X_t$ follows the flow until the next jump time $T_2 = T_1 + S_2$

$$X_{T_1 + t} = (M_1, \phi_{M_1}(Y_1, t)), \quad t < S_2$$
Iterative construction

Post-jump location \( Z_2 = (M_2, Y_2) \) selected by Markov kernel

\[ Q_{M_1} (\phi_{M_1} (Y_1, S_2), \cdot) \ldots \]
Embedded Markov chain

\{X_t\} strong Markov process [Davis 93]

Natural embedded Markov chain

- $Z_0$ starting point, $T_0 = 0$
- $Z_n$ new mode and location after $n$-th jump
- $T_n$ date of $n$-th jump

Proposition

$(Z_n, T_n)$ is a discrete-time Markov chain
Only source of randomness of the PDMP
Outline

Motivation

Piecewise deterministic Markov processes

Exit time problem
  Recursive formulation
  Discretization of the Markov chain
  Approximation scheme and error derivation

Numerical results

Conclusion
Exit time problem

$U$ open subset of $E$. Exit time from $U$

$$\tau = \inf\{s \geq 0 : X_s \notin U\}$$

$X_0 \in U$, $\mathbb{P}(\tau < \infty) = 1$

**Aim**

Propose a general numerical method to approximate the law of $\tau$

$$s \mapsto \mathbb{P}(\tau > s)$$
Recursive formulation

\[ \{ \tau \leq T_{k+1} \} = \{ \tau \leq T_k \} \cup \{ T_k < \tau \leq T_{k+1} \} \]

Consequence: recursive computation of \( \mathbb{P}(\tau > s | \tau \leq T_n) \)

\[
\mathbb{P}(\tau > s | \tau \leq T_{k+1}) = \frac{\mathbb{P}(\tau > s | \tau \leq T_k) \mathbb{P}(\tau \leq T_k) + \mathbb{P}(\{ \tau > s \} \cap \{ T_k < \tau \leq T_{k+1} \})}{\mathbb{P}(\tau \leq T_{k+1})}
\]

\( \mathbb{P}(\tau > s | \tau \leq T_0) = 0 \)
New formulation

Compute two quantities

- $\mathbb{P}(\tau \leq T_k)$
- $\mathbb{P}(\{\tau > s\} \cap \{T_k < \tau \leq T_{k+1}\})$

Strategy
New formulation

Compute two quantities

- \( \mathbb{P}(\tau \leq T_k) \)
- \( \mathbb{P}(\{\tau > s\} \cap \{T_k < \tau \leq T_{k+1}\}) \)

Strategy

- make appear the Markov chain \((Z_n, T_n)\)
New formulation

Compute two quantities

- $P(\tau \leq T_k)$
- $P(\{\tau > s\} \cap \{T_k < \tau \leq T_{k+1}\})$

Strategy

- make appear the Markov chain $(Z_n, T_n)$
- Discretize la chaîne de Markov $(Z_n, T_n)$
Formulation with the embedded Markov chain

- \( \mathbb{P}(\tau \leq T_k) \)

- \( \mathbb{P}(\{\tau > s\} \cap \{T_k < \tau \leq T_{k+1}\}) \)
Formulation with the embedded Markov chain

- $\mathbb{P}(\tau \leq T_k) = \mathbb{E}[1_{U^c}(Z_k)]$
  provided the process does not return to $U$ after the first exit
  $\rightarrow$ kill the process when it exits $U$

- $\mathbb{P}(\{\tau > s\} \cap \{T_k < \tau \leq T_{k+1}\})$
Formulation with the embedded Markov chain

- $\mathbb{P}(\tau \leq T_k) = \mathbb{E}[\mathbb{1}_{U^c}(Z_k)]$
  provided the process does not return to $U$ after the first exit
  $\longrightarrow$ kill the process when it exits $U$

- $\mathbb{P}(\{\tau > s\} \cap \{T_k < \tau \leq T_{k+1}\})$
  $= \mathbb{E}[\mathbb{1}_U(Z_k)\mathbb{1}_{U^c}(Z_{k+1})]$
Formulation with the embedded Markov chain

\[ P(\tau \leq T_k) = \mathbb{E}[\mathbb{1}_{U^c}(Z_k)] \]
provided the process does not return to \( U \) after the first exit
\[ \longrightarrow \text{kill the process when it exits } U \]

\[ P(\{\tau > s\} \cap \{T_k < \tau \leq T_{k+1}\}) = \mathbb{E}\left[ \mathbb{1}_{ \{(T_k+u^*(Z_k)) \wedge T_{k+1} > s\} } \mathbb{1}_U(Z_k) \mathbb{1}_{U^c}(Z_{k+1}) \right] \]
Formulation with the embedded Markov chain

- \( \mathbb{P}(\tau \leq T_k) = \mathbb{E}[1_{U^c}(Z_k)] \)
  provided the process does not return to \( U \) after the first exit
  \( \rightarrow \) kill the process when it exits \( U \)

- \( \mathbb{P}(\{\tau > s\} \cap \{T_k < \tau \leq T_{k+1}\}) \)
  \[= \mathbb{E}[1_{\{T_k + u^*(Z_k) \land T_{k+1} > s\}} 1_U(Z_k) 1_{U^c}(Z_{k+1})] \]
Quantization

Quantization of a random variable $X \in L^p(\mathbb{R}^d)$

Approximate $X$ by $\hat{X}$ taking **finitely** many values such that $\|X - \hat{X}\|_p$ is **minimum**

- Find a finite weighted grid $\Gamma$ with $|\Gamma| = K$
- Set $\hat{X} = p_\Gamma(X)$ closest neighbor projection

Asymptotic properties

If $E[|X|^{p+\eta}] < +\infty$ for some $\eta > 0$ then

$$\lim_{K \to \infty} K^{1/d} \min_{|\Gamma| \leq K} \|X - \hat{X}_\Gamma\|_p = C$$
Algorithms

There exist algorithms providing
- $\Gamma$
- law of $\hat{X}$
- transition probabilities for quantization of Markov chains

Example: $\mathcal{N}(0, I_2)$:
Algorithms

There exist algorithms providing

- $\Gamma$
- law of $\hat{X}$
- transition probabilities for quantization of Markov chains

Example: $\mathcal{N}(0, I_2)$:
Algorithms

There exist algorithms providing

- $\Gamma$
- law of $\hat{X}$
- transition probabilities for quantization of Markov chains

Example: $\mathcal{N}(0, I_2)$:
Grids construction

Model $\rightarrow$ simulator of trajectories $\rightarrow$ grids
Grids construction

Model $\rightarrow$ simulator of trajectories $\rightarrow$ grids
Grids construction

Model $\rightarrow$ simulator of trajectories $\rightarrow$ grids
Grids construction

Model $\rightarrow$ simulator of trajectories $\rightarrow$ grids
Assets and drawbacks of quantization

**Assets**

- a simulator of the target law is enough to build the grids
- automatic construction of grids
- convergence rate for $\mathbb{E}[|f(X) - f(\hat{X})|]$ if $f$ lipschitz

**Drawbacks**

- computation time
- curse of dimension
- open questions of convergence of the algorithms
Numerical scheme

Replace \((Z_n, T_n)\) by its quantized approximation \((\hat{Z}_n, \hat{T}_n)\)

\[
\hat{P}(\tau > s | \tau \leq T_0) = 0
\]

\[
\hat{P}(\tau > s | \tau \leq T_{k+1}) = \frac{1}{\mathbb{E}[\mathbb{1} U_c(\hat{Z}_{k+1})]} \left( \hat{P}(\tau > s | \tau \leq T_k) \mathbb{E}[\mathbb{1} U_c(\hat{Z}_k)] + \mathbb{E}[\mathbb{1} \{ \hat{T}_k + u^*(\hat{Z}_k) \land \hat{T}_{k+1} > s \} \mathbb{1} U(\hat{Z}_k) \mathbb{1} U_c(\hat{Z}_{k+1})] \right)
\]

if \(\mathbb{E}[\mathbb{1} U_c(\hat{Z}_{k+1})] \neq 0\), \(\hat{P}(\tau > s | \tau \leq T_{k+1}) = 0\) otherwise
Error derivation: dealing with indicator functions

\[ \mathbb{E}[|1_U(Z_k) - 1_U(\hat{Z}_k)|] \] small?

\[ \begin{align*}
\leq & \phantom{+} 0 \text{ if } Z_k \text{ and } \hat{Z}_k \text{ are on the same side of } \partial U \\
\leq & \mathbb{P}(|Z_k - \hat{Z}_k| > \epsilon) + \mathbb{P}(d(Z_k, \partial U) \leq \epsilon) \text{ otherwise}
\end{align*} \]
Error derivation: dealing with indicator functions

\[ \mathbb{E}[|\mathbbm{1}_U(Z_k) - \mathbbm{1}_U(\hat{Z}_k)|] \text{ small ?} \]

- \( = 0 \) if \( Z_k \) and \( \hat{Z}_k \) are on the same side of \( \partial U \)
- \( \leq \mathbb{P}(|Z_k - \hat{Z}_k| > \epsilon) + \mathbb{P}(d(Z_k, \partial U) \leq \epsilon) \) otherwise

\[ \mathbb{P}(d(Z_k, \partial U) \leq \epsilon) \leq C\epsilon^\beta \]

Control the probability to have a post-jump location close to the boundary of \( U \)
Error derivation: dealing with the inverse function

**Lipschitz continuity assumption on the parameters of the PDMP**

### Main other difficulties

- **Recurrence relation not Lipschitz** because of denominator
  
  - prove that $\mathbb{E}[1_{U^c}(\hat{Z}_{k+1})]$ is either 0 or $\geq q > 0$

- **$Z_k \in U$ does not generally imply $\hat{Z}_k \in U$**
  
  - $U$ convex
Convergence result

**Theorem**

For all $s \geq 0$ and $k$ smaller than the horizon $N$

$$\left| \hat{P}(\tau > s \mid \tau \leq T_k) - P(\tau > s \mid \tau \leq T_k) \right| \longrightarrow 0$$

when the quantization error goes to 0
Outline

Motivation

Piecewise deterministic Markov processes

Exit time problem

Numerical results
  Quantization grids
  Service time

Conclusion
Practical problem for the corrosion model quantization

Parameters with different scales

- corrosion rate $\rho \sim 10^{-6} \text{ mm/h}$
- mean time spent in the submarine 2: $\lambda_2^{-1} = 131\,400 \text{ h}$
Practical problem for the corrosion model quantization

Parameters with different scales

- corrosion rate $\rho \sim 10^{-6}$ mm/h
- mean time spent in the submarine 2: $\lambda_2^{-1} = 131 400$ h

Uniform law on $[0, 1] \times [0, 5000]$
Practical problem for the corrosion model quantization

Parameters with different scales

- corrosion rate $\rho \sim 10^{-6} \text{ mm/h}$
- mean time spent in the submarine 2: $\lambda_2^{-1} = 131 400 \text{ h}$

Uniform law on $[0, 1] \times [0, 5000]$

weighted algorithm
Quantization grids for the corrosion process

Environment 2 after jump 1
Quantization grids for the corrosion process

Environment 3 after jump 2
Quantization grids for the corrosion process

Environment 1 after jump 15
Numerical computation of the service time

\( U = [0, 0.2] mm, \ \tau = \text{service time of the structure} \)

<table>
<thead>
<tr>
<th>Quantization grids</th>
<th>max of the error</th>
<th>max of relative error</th>
</tr>
</thead>
<tbody>
<tr>
<td>20 points</td>
<td>0.1041</td>
<td>15%</td>
</tr>
<tr>
<td>50 points</td>
<td>0.0664</td>
<td>13%</td>
</tr>
<tr>
<td>100 points</td>
<td>0.0447</td>
<td>9.3%</td>
</tr>
<tr>
<td>200 points</td>
<td>0.0335</td>
<td>8%</td>
</tr>
<tr>
<td>500 points</td>
<td>0.0112</td>
<td>5.8%</td>
</tr>
<tr>
<td>1000 points</td>
<td>0.0091</td>
<td>4.6%</td>
</tr>
<tr>
<td>2000 points</td>
<td>0.0058</td>
<td>3.6%</td>
</tr>
<tr>
<td>4000 points</td>
<td>0.0039</td>
<td>2.8%</td>
</tr>
<tr>
<td>6000 points</td>
<td>0.0028</td>
<td>2.2%</td>
</tr>
<tr>
<td>8000 points</td>
<td>0.0024</td>
<td>2.0%</td>
</tr>
<tr>
<td>10000 points</td>
<td>0.0023</td>
<td>1.8%</td>
</tr>
</tbody>
</table>
Approximate service time

![Graph showing approximate service time with 8000 points and Monte Carlo methods.]
Empirical error

slope $-0.35 \leftarrow$ quantization error for 3-dimensional process (thickness loss, corrosion rate, time)
# Assets and drawbacks

<table>
<thead>
<tr>
<th>Drawbacks</th>
</tr>
</thead>
<tbody>
<tr>
<td>- not competitive with Monte Carlo simulation if <em>only one</em> computation is needed</td>
</tr>
<tr>
<td>- drawback of <em>quantization</em>: dimension, time to compute the grids</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Assets</th>
</tr>
</thead>
<tbody>
<tr>
<td>- <em>same grids</em> for different computations, do not change if $U$ changes</td>
</tr>
<tr>
<td>- <em>fast</em> on-line computation once the grids are pre-computed</td>
</tr>
</tbody>
</table>
Thank you