Numerical method for impulse control of Piecewise Deterministic Markov Processes

Benoîte de Saporta, François Dufour, Huilong Zhang
Univ. Montpellier, Bordeaux INP, Univ. Bordeaux

Outline

Maintenance optimization

Equipments

- with several components
- subject to random failures

Choose the best interventions

- when ?
- what type : change or repair ?

In order to optimize some criterion

- minimize a cost: functioning, maintenance, ...
- maximize a reward: availability, ...

Our approach

- propose a general model for the evolution of the equipment state based on PDMPs
- formalize the maintenance problem as an impulse control problem for PDMPs
- derive a numerical scheme to approximate the value function (with error bounds)
- compute the approximate optimal maintenance cost

Piecewise deterministic Markov processes

[Davis 93] General class of non-diffusion dynamic stochastic hybrid models: deterministic motion punctuated by random jumps.

Hybrid process $X_t = (m_t, x_t)$

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

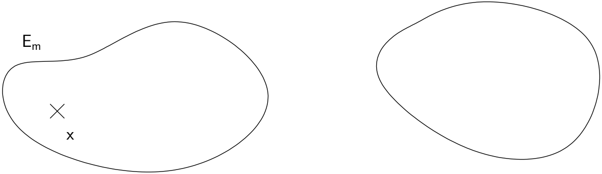
Local characteristics for each mode m

- E_m open subset of $\mathbb{R}^d$
- Flow ϕ_m: $\mathbb{R}^d \times \mathbb{R} \to \mathbb{R}^d$ deterministic motion between jumps, one-parameter group of homeomorphisms
- Intensity λ_m: $\overline{E}_m \to \mathbb{R}_+$ intensity of random jumps
- Markov kernel Q_m on $(\overline{E}_m, \mathcal{B}(\overline{E}_m))$ selects the post-jump location

Dynamics

Starting point

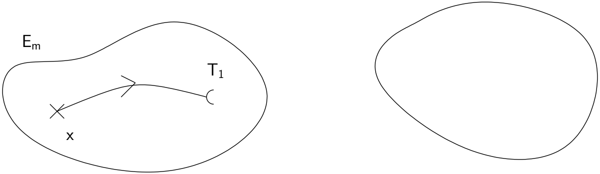
$$X_0 = (m, x)$$



Dynamics

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

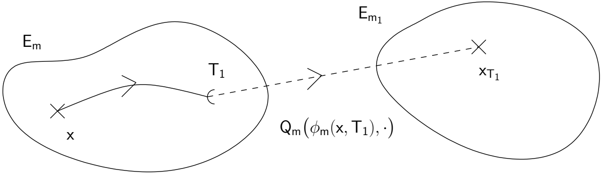
$$X_t = \big(m, \phi_m(x,t)\big), \quad \mathbb{P}_{(m,x)}(S_1 > t) = \mathrm{e}^{-\int_0^t \lambda_m\left(\phi_m(x,s)\right) ds}$$



Dynamics

Post-jump location (m_1, x_{T_1}) selected by the Markov kernel

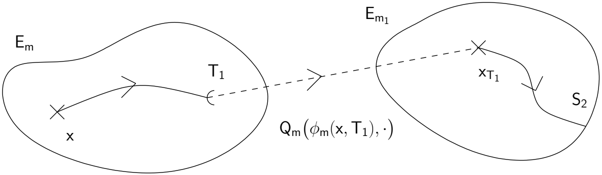
$$Q_m(\phi_m(x, T_1), \cdot)$$



Dynamics

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

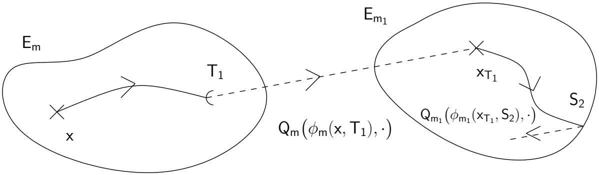
$$X_{T_1+t} = \big(m_1, \phi_{m_1}(x_{T_1}, t)\big), \quad t < S_2$$



Dynamics

Post-jump location (m_2, x_{T_2}) selected by Markov kernel

$$Q_{m_1}(\phi_{m_1}(x_{T_1}, S_2), \cdot) \ldots$$



Embedded Markov chain

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

Natural embedded Markov chain

- Z_0 starting point, $S_0 = 0$, $S_1 = T_1$
- Z_n new mode and location after n-th jump, $S_n = T_n - T_{n-1}$, time between two jumps

Proposition

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

Equipement model

Typical model with 4 components

- Component 1: 2 states – stable $\xrightarrow{\text{Exponential}}$ failed
- Component 2: 2 states – stable $\xrightarrow{\text{Weibull}}$ failed
- Components 3 and 4: 3 states

  stable $\xrightarrow{\text{Weibull}}$ degraded $\xrightarrow{\text{Exponential}}$ failed

Equipement model

Typical model with 4 components

- Component 1: 2 states – stable $\xrightarrow{\text{Exponential}}$ failed
- Component 2: 2 states – stable $\xrightarrow{\text{Weibull}}$ failed
- Components 3 and 4: 3 states
  stable $\xrightarrow{\text{Weibull}}$ degraded $\xrightarrow{\text{Exponential}}$ failed

Possible maintenance operations

- Components 1 and 2: change
- Components 3 and 4: change , repair (only in stable or degraded states)

Global equipment state

The equipment is globally

- stable if the 4 components are stable
- degraded if at leat one component is degraded and the others are stable or degraded
- failed if at least one component is failed failed
- int the workshop if there is an ongoing maintenant operation of change or repair

PDMP model of the equipment

- Euclidean variables
  - functioning time of components 2, 3 and 4
  - calendar time
  - time spent in the workshop
- Discrete variables
  - state of the components / maintenance operations

Other applications of PDMPs

Applications of PDMPs

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

- mode: nominal, failures, breakdown, environment, number of individuals, response to a treatment, . . .
- Euclidean variable: pressure, temperature, time, size, potential, protein level, . . .

Impulse control problem

Impulse control

Select

- intervention dates
- new starting point for the process at interventions

to minimize a cost function

- repair a component before failure
- change treatment before relapse
- ...

[CD 89], [Davis 93], [dSDZ 14], ...

Mathematical definition

Strategy $\mathcal{S} = (\tau_n, R_n)_{n \geq 1}$

- τ_n intervention times
- R_n new positions after intervention

Value function

$$\mathcal{J}^{\mathcal{S}}(x) = E_x^{\mathcal{S}}\left[\int_0^{\infty} e^{-\alpha s} f(Y_s) ds + \sum_{i=1}^{\infty} e^{-\alpha \tau_i} c(Y_{\tau_i}, Y_{\tau_i^+})\right]$$

$$\mathcal{V}(x) = \inf_{\mathcal{S} \in \mathbb{S}} \mathcal{J}^{\mathcal{S}}(x)$$

- f, c cost functions, α discount factor
- Y_t controlled process, $\mathbb{S}$ set of admissible strategies

Example of maintenance optimization

- τ_n: maintenance dates
- R_n: which components are to be changed/repaired

Value function

$$\mathcal{J}^{\mathcal{S}}(x) = E_x^{\mathcal{S}}\left[\int_0^\infty e^{-\alpha s} f(Y_s) ds + \sum_{i=1}^{\infty} e^{-\alpha \tau_i} c(Y_{\tau_i}, Y_{\tau_i^+})\right]$$

$$\mathcal{V}(x) = \inf_{\mathcal{S} \in \mathbb{S}} \mathcal{J}^{\mathcal{S}}(x)$$

- f unavailability cost proportional to time spend in failed state
- c fixed cost for going to the workshop + repair $<$ change costs
- $\alpha = 0$ (finite horizon)

Dynamic programming

Costa, Davis, 1988

For any function $g \geq$ cost of the no-impulse strategy

- $v_0 = g$
- $v_n = \mathcal{L}(v_{n-1})$

$$v_n(x) \xrightarrow[n\to\infty]{} \mathcal{V}(x)$$

Dynamic programming operator

$$
\begin{aligned}
\mathcal{L}(w)(x) &= L(Mw, w)(x) \\
&= \Big(\inf_{t \le t^*(x)} \mathbb{E}_x \Big[F(x,t) + \mathrm{e}^{-\alpha S_1} w(Z_1) \mathbb{1}_{\{S_1 < t \wedge t^*(x)\}} \\
&\quad + \mathrm{e}^{-\alpha t \wedge t^*(x)} Mw(\phi(x, t \wedge t^*(x)) \mathbb{1}_{\{S_1 \ge t \wedge t^*(x)\}} \Big] \Big) \\
&\quad \wedge \mathbb{E}_x \Big[F(x, t^*(x)) + \mathrm{e}^{-\alpha S_1} w(Z_1) \Big]
\end{aligned}
$$

with

$$
\begin{aligned}
F(x,t) &= \int_0^{t \wedge t^*(x)} \mathrm{e}^{-\alpha s - \Lambda(x,s)} f\big(\phi(x,s)\big) ds \\
Mw(x) &= \inf_{y \in \mathbb{U}} \big\{ c(x,y) + w(y) \big\}
\end{aligned}
$$

Our aim

Propose a numerical method

- to compute an approximation of the value function
- with error bounds

Main difficulty

Discretization of the dynamic programming operator

Our approach

Discretize the underlying Markov chain (Z_n, S_n) by optimal quantization

Quantization

Quantization of a random variable X

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

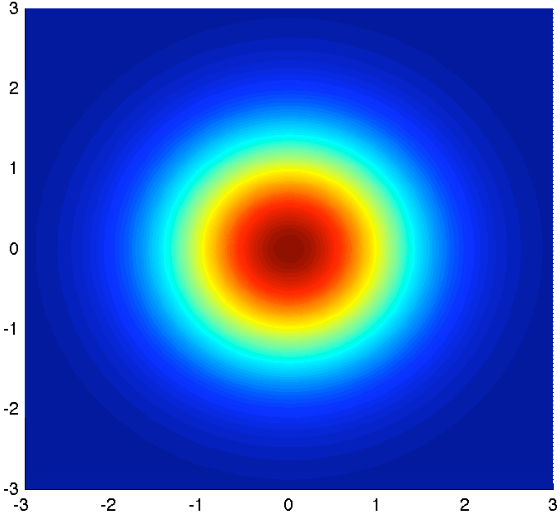
- Find a finite weighted grid Γ with $|\Gamma| = K$
- Set $\widehat{X} = p_\Gamma(X)$ closest neighbor projection

Algorithms

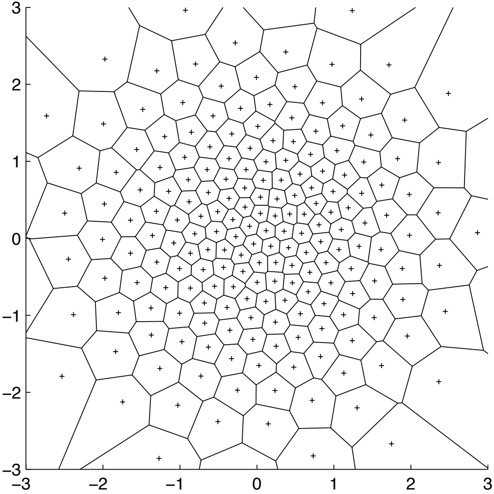
There exist algorithms providing

- Γ
- law of $\widehat{X}$
- transition probabilities for quantization of Markov chains

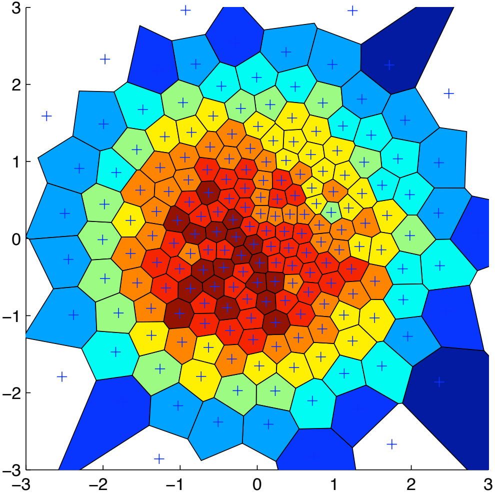
Example: $\mathcal{N}(0, I_2)$:



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



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



Horizon and control set

- finite set $\mathbb{U}$ of new starting points
- select horizon N such that $v_N(x) - \mathcal{V}(x)$ small enough

$\longrightarrow$ numerical approximation of $v_N(x)$

Main idea

Replace the dynamic programming iteration of functions by an iteration of random variables

Backward dynamic programming

Change of notation

For a well chosen function g and large enough N

- $v_N = g$
- $v_n = \mathcal{L}(v_{n+1})$

$$v_0(x) \simeq \mathcal{V}(x)$$

Dynamic programming

Markov property

$$
\begin{aligned}
v_n(Z_n) &= \mathcal{L}(Mv_{n+1}, v_{n+1})(Z_n) \\
&= \Big(\inf_{t \le t^*(Z_n)} \mathbb{E}\Big[F(Z_n, t) + \mathrm{e}^{-\alpha S_{n+1}} v_{n+1}(Z_{n+1}) 1_{\{S_{n+1} < t \wedge t^*(Z_n)\}} \\
&\quad + \mathrm{e}^{-\alpha t \wedge t^*(Z_n)} Mv_{n+1}\big(\phi(Z_n, t \wedge t^*(Z_n))\big) \mathbb{1}_{\{S_{n+1} \ge t \wedge t^*(Z_n)\}} \mid Z_n \Big] \\
&\quad \wedge \mathbb{E}\Big[F(Z_n, t^*(Z_n)) + \mathrm{e}^{-\alpha S_{n+1}} v_{n+1}(Z_{n+1}) \mid Z_n \Big]
\end{aligned}
$$

with

$$
\begin{aligned}
F(x, t) &= \int_0^{t \wedge t^*(x)} \mathrm{e}^{-\alpha s - \Lambda(x,s)} f\big(\phi(x, s)\big) ds \\
Mv_{n+1}(x) &= \inf_{y \in \mathbb{U}} \big\{ c(x, y) + v_{n+1}(y) \big\}
\end{aligned}
$$

Recursion on random variables

$v_n(Z_n)$ expression of $v_{n+1}(Z_{n+1})$, Z_n, S_{n+1}
$+\ v_{n+1}(y)$ for all y in $\mathbb{U}$

Numerical scheme

- first compute recursively $\widetilde{v}_n(y)$ approximation of $v_n(y)$ for all y in $\mathbb{U}$
- then compute recursively $\widehat{v}_n(\widehat{Z}_n)$ approximation of $v_n(Z_n)$

Discretization

In the expression of operator $\mathcal{L}$ replace

- inf by min over a discretized grid
- Z_n, Z_{n+1}, S_{n+1} by their quantized approximation starting from $Z_0 \in \mathbb{U}$

Recursion on random variables

$v_n(Z_n)$ expression of $v_{n+1}(Z_{n+1})$, Z_n, S_{n+1}
$+ \ v_{n+1}(y)$ for all y in $\mathbb{U}$

Numerical scheme

- first compute recursively $\widetilde{v}_n(y)$ approximation of $v_n(y)$ for all y in $\mathbb{U}$
- then compute recursively $\widehat{v}_n(\widehat{Z}_n)$ approximation of $v_n(Z_n)$

Discretization

In the expression of operator $\mathcal{L}$ replace

- inf by min over a discretized grid
- Z_n, Z_{n+1}, S_{n+1} by their quantized approximation starting from $Z_0 = x$

Properties of the numerical scheme

- the quantized process has no Markov property $\rightarrow$ a different approximation of $\mathcal{L}$ for each time step and each starting point
- Under Lipschitz regularity assumption, convergence of the scheme with errors bounds depending on
  - the time discretization step inf $\rightarrow$ min
  - the quantization error $(Z_n, S_n) \rightarrow (\widehat{Z}_n, \widehat{S}_n)$

Step 1: Exact simulation of the PDMP

Implementation of an exact simulator for reference strategies to serve as benchmark

	1	2	3	4	5
intervention	never	1 day failed	1 day failed	1 day degraded or failed	1 day degraded or failed
C1 failed	nothing	change	change	change	change
C3 degraded	nothing	change	repair	change	repair
C3 failed	nothing	change	change	change	change
C2 failed and C4 stable	nothing	change 2+4	change 2+4	change 2+4	2+4 2+4
C2 failed and C4 degraded	nothing	change 2+4	change 2+4	change 2+4	change 2+4
C2 stable and C4 degraded	nothing	change 2+4	repair 4	change 2+4	repair 4
C2 stable and C4 failed	nothing	change 2+4	change 2+4	change 2+4	change 2+4
Mean cost	19680	11184	11114	11521	8359

Step 2 : Discretisation of the control st $\mathbb{U}$

Finite control set $\mathbb{U}$
$\Longrightarrow$ discretize the functioning times at interventions
$\Longrightarrow$ project the real times on the grid feasibly

Compromise between precision and computation time

Tests on strategy 5

Grid	Number of points	relative error
$3 \times 3 \times 3 \times 5$	419	0.1458
$4 \times 4 \times 4 \times 5$	627	0.1331
$5 \times 5 \times 5 \times 5$	1055	0.1235
$3 \times 3 \times 3 \times 11$	788	0.0962
$4 \times 4 \times 4 \times 11$	1219	0.0819
$5 \times 5 \times 5 \times 11$	1855	0.0730
$6 \times 6 \times 6 \times 11$	2790	0.0672
$7 \times 7 \times 7 \times 11$	3570	0.0634
$8 \times 8 \times 8 \times 11$	4647	0.0604
$3 \times 3 \times 3 \times 21$	1403	0.0775
$4 \times 4 \times 4 \times 21$	2195	0.0626
$5 \times 5 \times 5 \times 21$	3423	0.0534
$6 \times 6 \times 6 \times 21$	4900	0.0436
$7 \times 7 \times 7 \times 21$	6489	0.0384
$8 \times 8 \times 8 \times 21$	8399	0.0350

Step 3: Discretizing the embedded Markov chain

- calibration on reference strategies

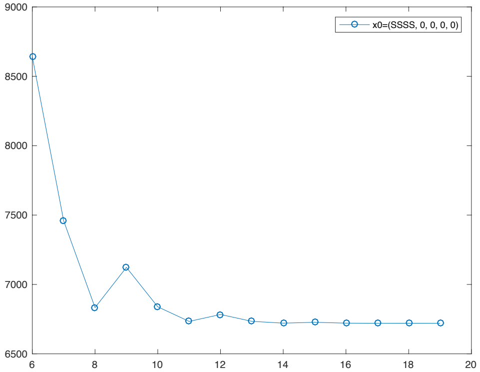
Compromise between precision and computation time

Number of points	Strategy 1	Strategy 2	Strategy 3	Strategy 4	Strategy 5
50	19680	11145	11075	11485	8326
100	19680	11207	11134	11509	8367
200	19680	11173	11104	11531	8361
400	19680	11193	11124	11531	8366
1000	19680	11180	11109	11517	8355
Exact cost	19680	11184	11114	11521	8359

Step 4: Calibrating N the number of allowed jumps + interventions

Horizon N (number of iterations)

- 5 for Strategy 1
- up to 30 for Strategies 2 and 3 (mean 6)
- up to 25 for Strategiess 4 and 5 (mean 6)



Step 5: Approximation of the value function

Strategy 1	Strategy 2	Strategy 3	Strategy 4	Strategy 5	Approx. Value function
19680	11184	11114	11521	8359	6720

- relative gain of 19.6% vs Strategy 5
- numerical validation of the algorithm with various starting points: consistent results

Conclusion and perspectives

Numerical method to approximate the value function

- rigorously validated
- with general error bounds
- numerical demanding but viable in low dimensional examples

Conclusion and perspectives

Numerical method to approximate the value function

- rigorously validated
- with general error bounds
- numerical demanding but viable in low dimensional examples

Work in progress

- Approximation of an ϵ-optimal strategy: numerical and theoretical study - PGMO grant 2018-2019

References

[CD 89] O. Costa, M. Davis *Impulse control of piecewise-deterministic processes*

[Davis 93] M. Davis, *Markov models and optimization*

[dSDG 17] B. de Saporta, F. Dufour, A. Geeraert *Optimal strategies for impulse control of piecewise deterministic Markov processes*

[dSDZ 14] B. de Saporta, F. Dufour, H. Zhang *Numerical methods for simulation and optimization of PDMPs: application to reliability*

[P 98] G. Pagès *A space quantization method for numerical integration*

[PPP 04] G. Pagès, H. Pham, J. Printems *An optimal Markovian quantization algorithm for multi-dimensional stochastic control problems*