



**HAL**  
open science

# Maintenance optimization with piecewise deterministic Markov processes

Benoîte de Saporta, François Dufour, Huilong Zhang

► **To cite this version:**

Benoîte de Saporta, François Dufour, Huilong Zhang. Maintenance optimization with piecewise deterministic Markov processes. Journées IMT-IMAG, Apr 2019, Toulouse, France. hal-02376724

**HAL Id: hal-02376724**

**<https://hal.science/hal-02376724>**

Submitted on 22 Nov 2019

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# Maintenance optimization with piecewise deterministic Markov processes

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



# Outline

Introduction

Motivation

Use case

Impulse control for PDMPs

Numerical implementation

Conclusion and perspectives

# Maintenance

From corrective actions to preventive and condition-based interventions

## Equipments

- ▶ with several components
- ▶ subject to **random** degradation and failures

**Maintenance policy**: sequence of interventions

- ▶ **when** ?
- ▶ what **type**: change or repair ?

Examples of maintenance policies

- ▶ change a component at failure
- ▶ repair or change a component every  $n$  months
- ▶ ...

# Maintenance optimization

From corrective actions to preventive and condition-based interventions

**Maintenance optimization problem:** find some optimal balance between

- ▶ repairing/changing components too often
- ▶ do nothing and wait for the total failure of the system

Optimize some criterion

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

# Maintenance optimization

From corrective actions to preventive and condition-based interventions

**Maintenance optimization problem:** find some optimal balance between

- ▶ repairing/changing components too often
- ▶ do nothing and wait for the total failure of the system

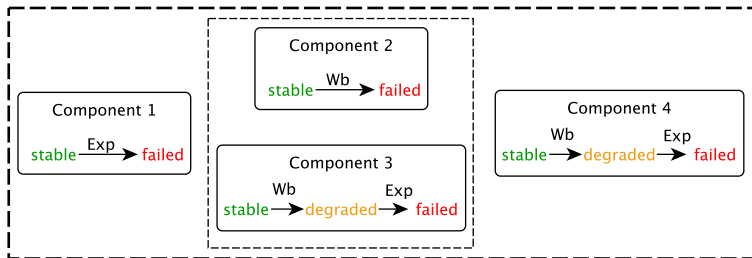
**Optimize** some criterion

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

# Equipment model

## Typical model with 4 components

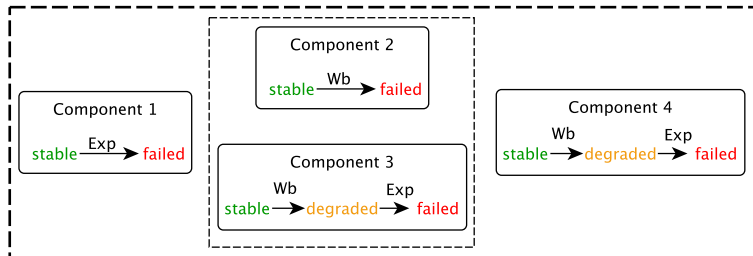
- ▶ Component 1: 2 states – **stable**  $\xrightarrow{\text{Exponential}}$  **failed**  $\mathbb{P}(T > t) = e^{-\lambda t}$
- ▶ Component 2: 2 states – **stable**  $\xrightarrow{\text{Weibull}}$  **failed**  $\mathbb{P}(T > t) = e^{-(t/\alpha)^\beta}$
- ▶ Components 3 and 4: 3 states  
**stable**  $\xrightarrow{\text{Weibull}}$  **degraded**  $\xrightarrow{\text{Exponential}}$  **failed**



# Maintenance operations

Possible **maintenance** operations

- ▶ All components, all states: do nothing
- ▶ Components 1 and 2, all states: **change**
- ▶ Components 3 and 4: **change** in all states, **repair** only in **stable** or **degraded** states

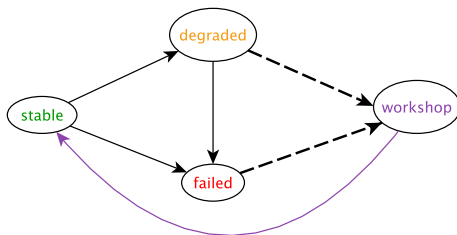




## Global state of the equipment

The equipment is globally

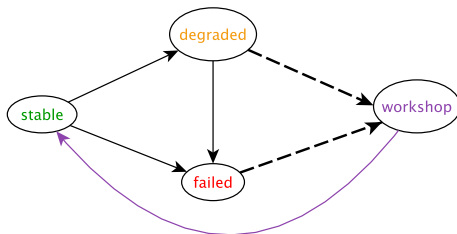
- ▶ **stable** if the 4 components are **stable**
- ▶ **degraded** if at least one component is **degraded** and the others are **stable** or **degraded**
- ▶ **failed** if at least one component is **failed**
- ▶ in the **workshop** if there is an ongoing maintenance operation of **change** or **repair**



# Criterion to optimize

Minimize the maintenance + unavailability costs

- ▶ **unavailability** cost proportional to time spend in **failed** state
- ▶ fixed cost for going to the workshop + **repair** < **change** costs



# 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
- ▶ propose a **computable** strategy close to optimality

# Outline

Introduction

Impulse control for PDMPs

- Piecewise deterministic Markov processes

- Optimization problem

- Discretization scheme

Numerical implementation

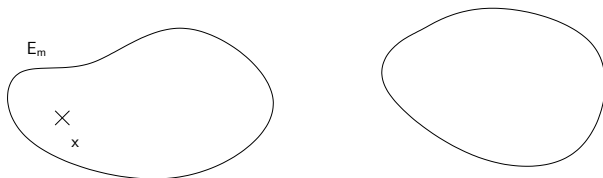
Conclusion and perspectives

# Piecewise deterministic Markov processes

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

Starting point

$$X_0 = (m, x)$$

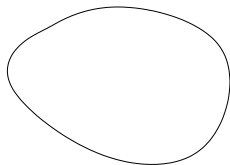
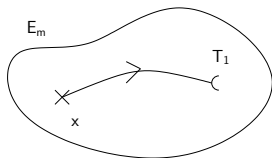


# Piecewise deterministic Markov processes

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

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

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

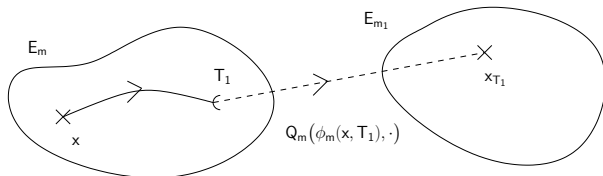


# Piecewise deterministic Markov processes

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

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

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

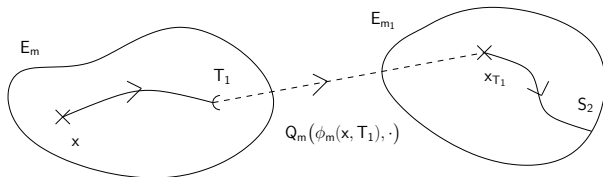


# Piecewise deterministic Markov processes

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

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

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



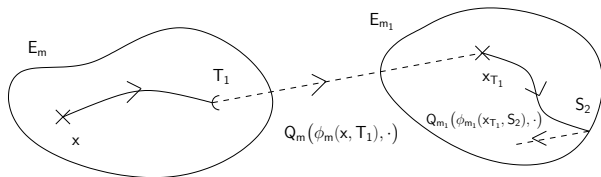


# Piecewise deterministic Markov processes

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

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

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



# 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

# Examples 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, ...

# PDMP model of the equipment

- ▶ **Euclidean variables:** 5 time variables
  - ▶ functioning time of components 2, 3 and 4
  - ▶ calendar time
  - ▶ time spent in the workshop
  
- ▶ **Discrete variables:** 225 modes
  - ▶ state of the components / maintenance operations

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

- ▶  $\tau_n$  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,  $\alpha$  discount factor
- ▶  $Y_t$  controlled process,  $\mathbb{S}$  set of admissible strategies

## Example of maintenance optimization

- ▶  $\tau_n$ : maintenance dates
- ▶  $R_n$ : which components are to be changed/repaired

### Value function

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

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

- ▶  $f$  **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

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

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

de Saporta, Dufour 2012

Numerical scheme to compute an approximation of the value function



# Dynamic programming operator

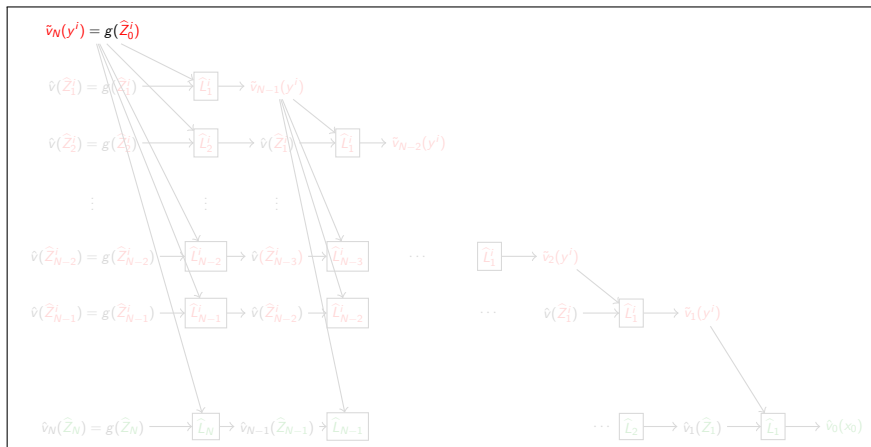
$$\begin{aligned}
 v_n(Z_n) &= \mathcal{L}(Mv_{n+1}, v_{n+1})(Z_n) \\
 &= \left( \inf_{t \leq t^*(Z_n)} \mathbb{E} \left[ F(Z_n, t) + e^{-\alpha S_{n+1}} v_{n+1}(Z_{n+1}) \mathbb{1}_{\{S_{n+1} < t \wedge t^*(Z_n)\}} \right. \right. \\
 &\quad \left. \left. + e^{-\alpha t \wedge t^*(Z_n)} Mv_{n+1}(\phi(Z_n,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

with

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

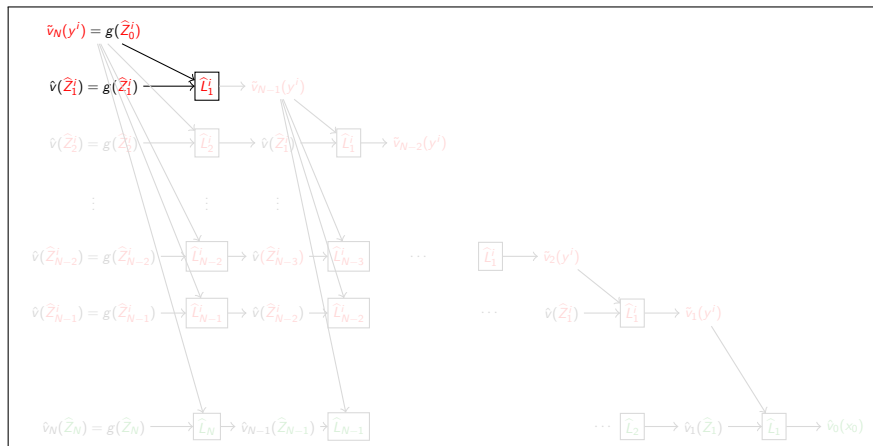
# Approximation scheme

## Value function



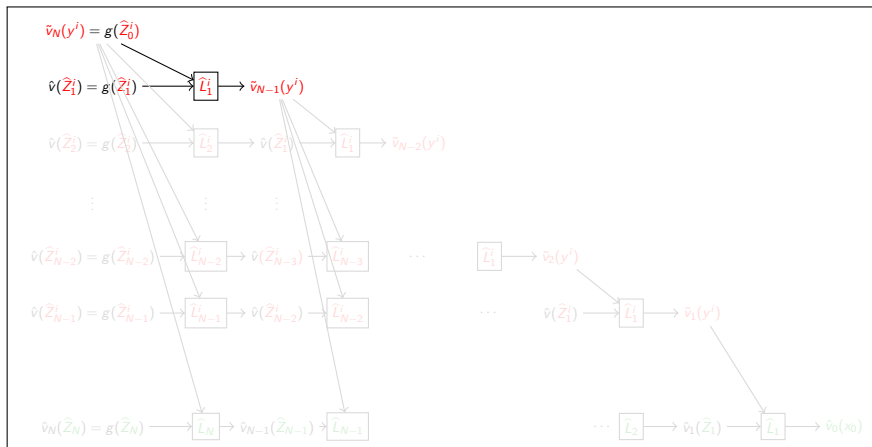
# Approximation scheme

## Value function



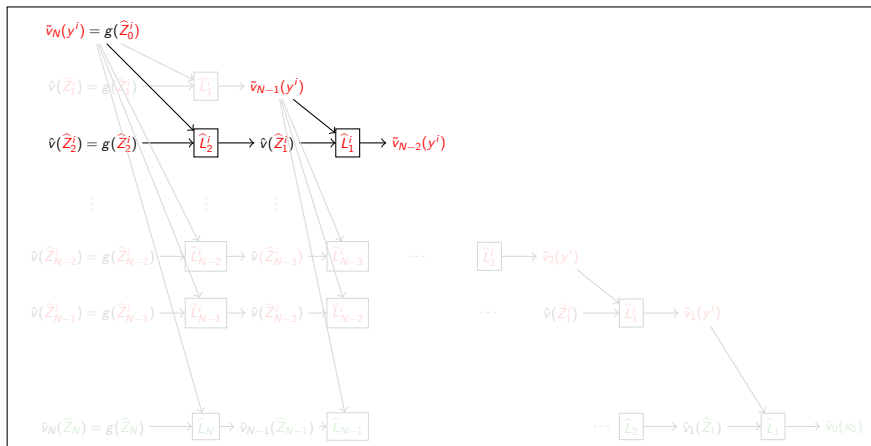
# Approximation scheme

## Value function



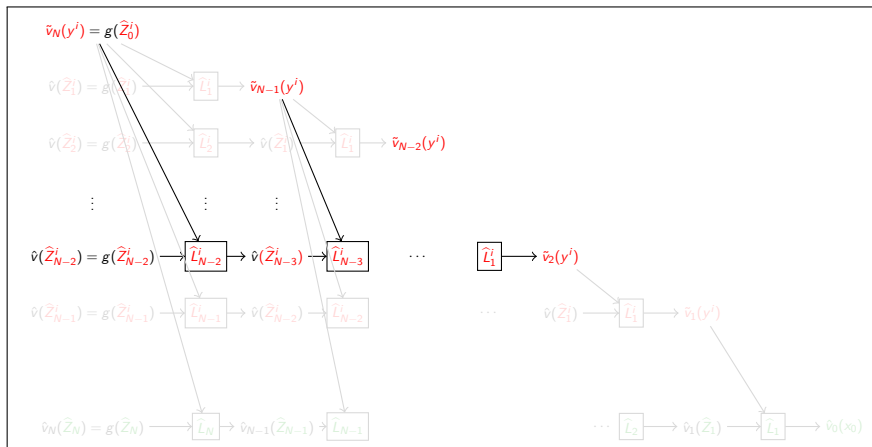
# Approximation scheme

## Value function



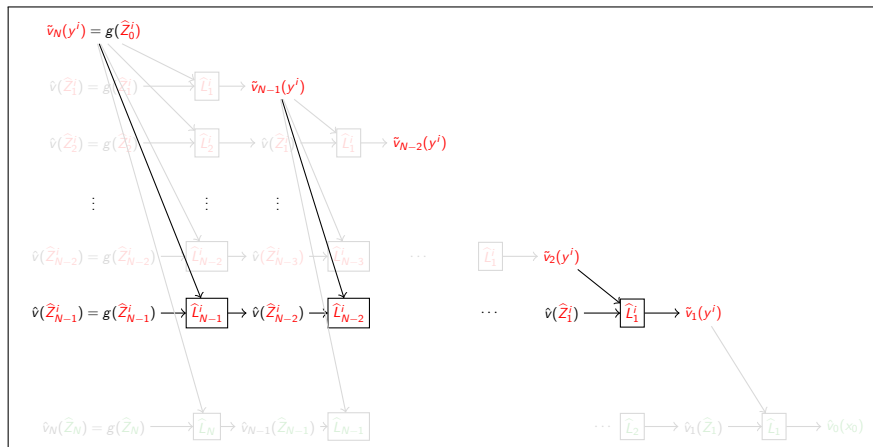
# Approximation scheme

## Value function



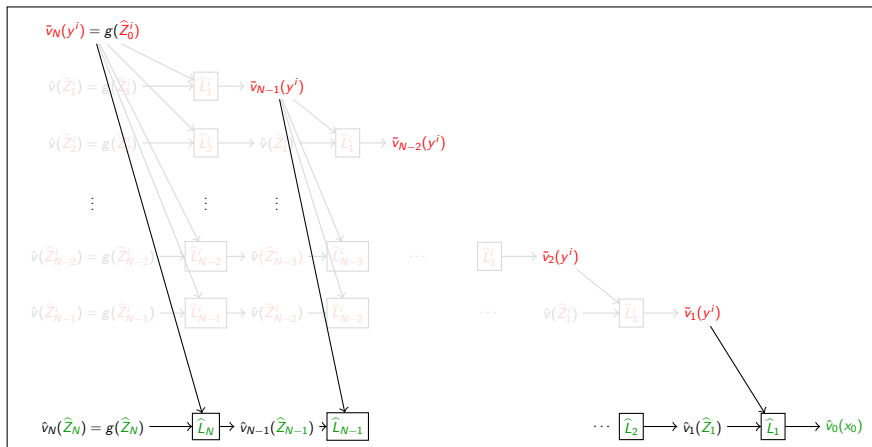
# Approximation scheme

## Value function



# Approximation scheme

## Value function





# $\epsilon$ -optimal strategy

## Dynamic programming operator

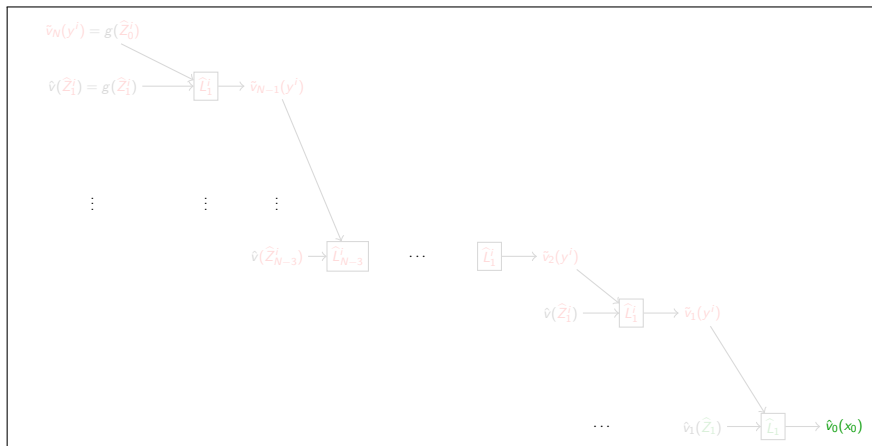
$$\begin{aligned}
 v_n(Z_n) &= \mathcal{L}(Mv_{n+1}, v_{n+1})(Z_n) \\
 &= \left( \inf_{t \leq t^*(Z_n)} \mathbb{E} \left[ F(Z_n, t) + e^{-\alpha S_{n+1}} v_{n+1}(Z_{n+1}) \mathbb{1}_{\{S_{n+1} < t \wedge t^*(Z_n)\}} \right. \right. \\
 &\quad \left. \left. + e^{-\alpha t \wedge t^*(Z_n)} Mv_{n+1}(\phi(Z_n,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

with

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

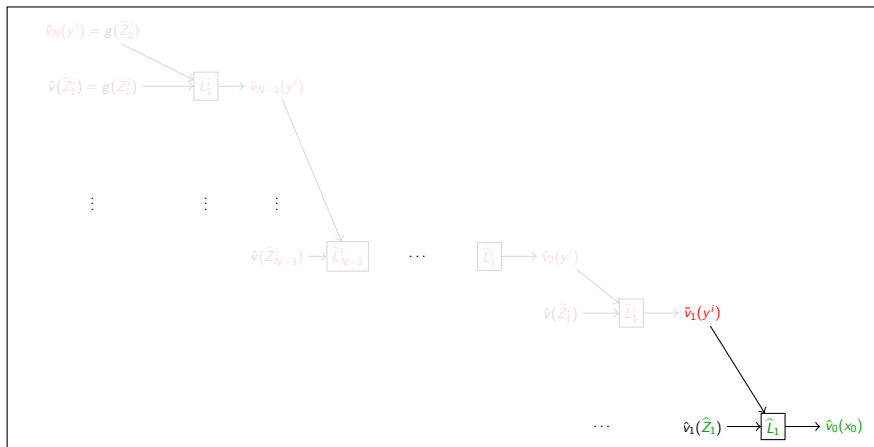
# Approximation scheme

$\epsilon$ -optimal strategy



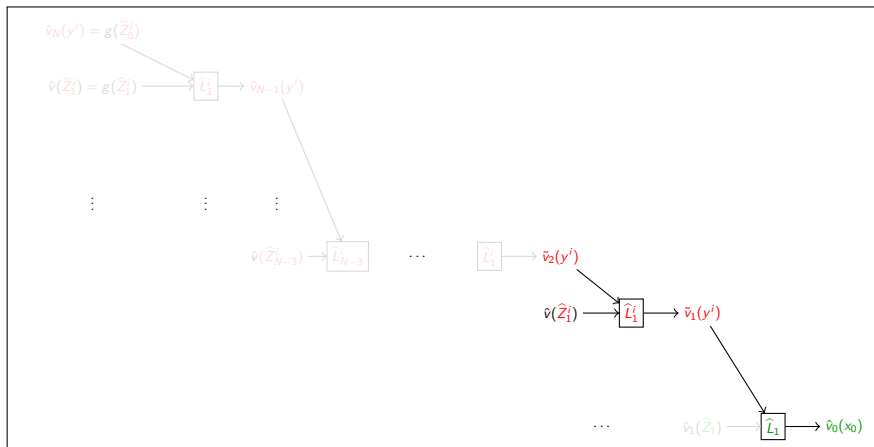
# Approximation scheme

$\epsilon$ -optimal strategy



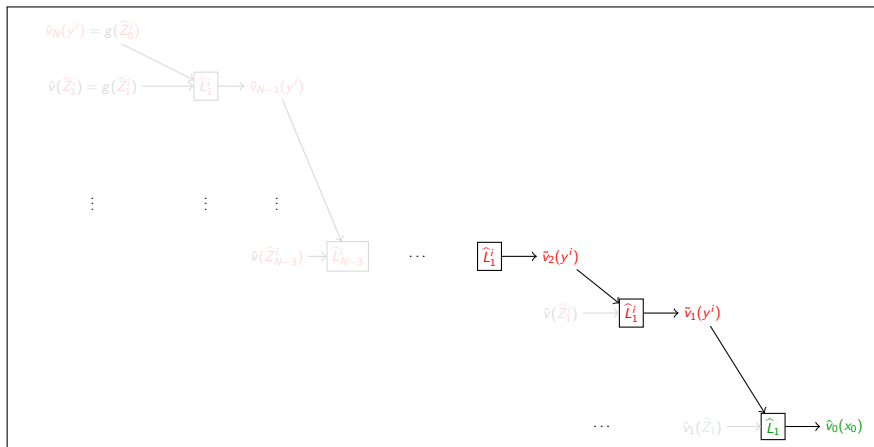
# Approximation scheme

$\epsilon$ -optimal strategy



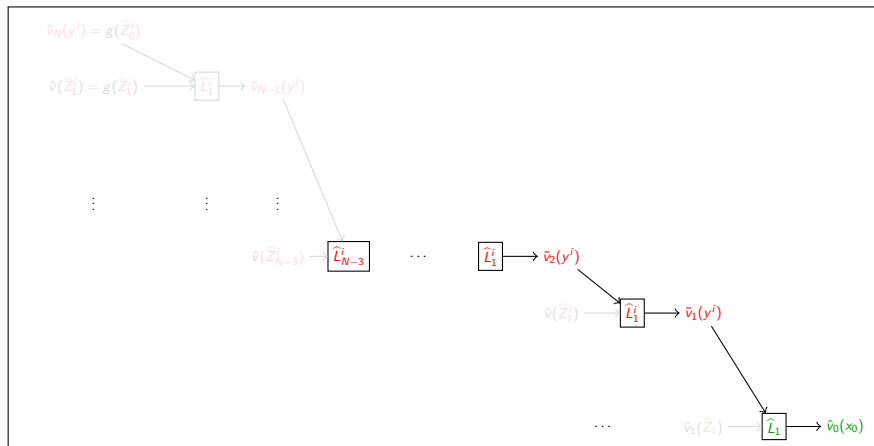
# Approximation scheme

$\epsilon$ -optimal strategy



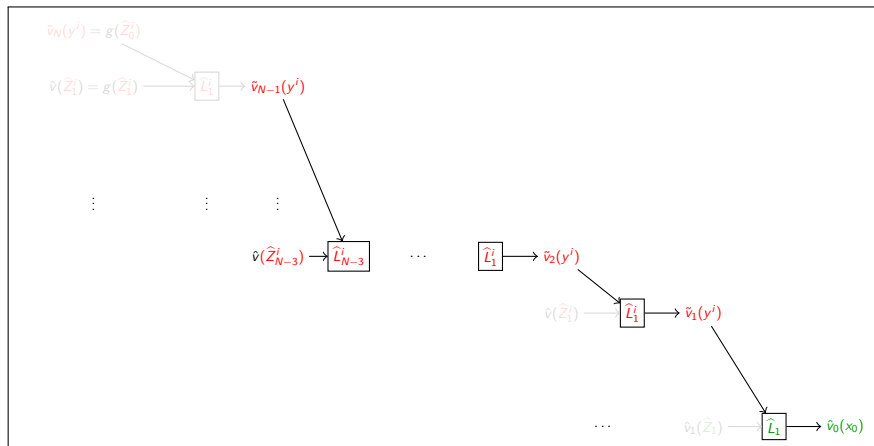
# Approximation scheme

$\epsilon$ -optimal strategy



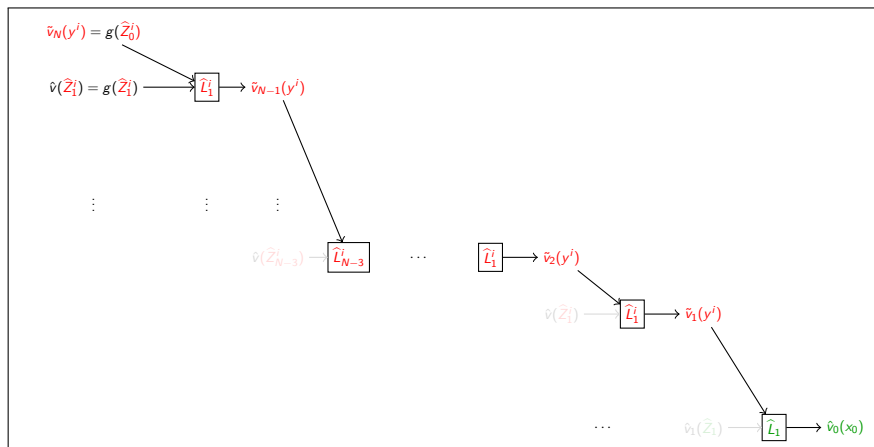
# Approximation scheme

$\epsilon$ -optimal strategy



# Approximation scheme

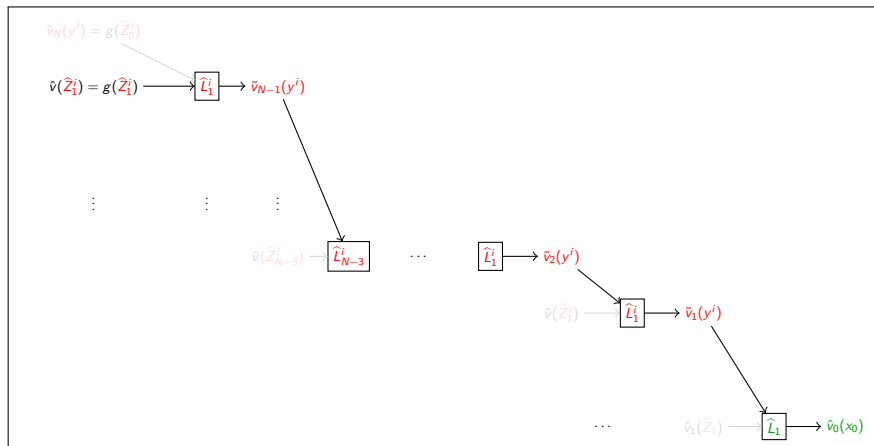
$\epsilon$ -optimal strategy





# Approximation scheme

$\epsilon$ -optimal strategy



# Outline

Introduction

Impulse control for PDMPs

**Numerical implementation**

Conclusion and perspectives

# Parameters to tune

- ▶ Number of points in the **control grid** (underlying continuous model)
- ▶ Number of point in the **quantization grids**
- ▶ **Approximation horizon**  $N$  such that  $v_N(x) - \mathcal{V}(x)$  small enough
- ▶ bounding function  $g$
- ▶ **Time discretization step** for inf

## Step 1: Exact simulation of the PDMP

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

|                              | 1       | 3               | 5                           |
|------------------------------|---------|-----------------|-----------------------------|
| intervention date            | never   | 1 day<br>failed | 1 day degraded<br>or failed |
| C1 failed                    | nothing | change          | change                      |
| C4 degraded                  | nothing | repair          | repair                      |
| C4 failed                    | nothing | change          | change                      |
| C2 failed<br>and C3 stable   | nothing | change<br>2+3   | change<br>2+3               |
| C2 failed<br>and C3 degraded | nothing | change<br>2+3   | change<br>2+3               |
| C2 stable<br>and C3 degraded | nothing | repair<br>3     | repair<br>3                 |
| C2 stable<br>and C3 failed   | nothing | change<br>2+3   | change<br>2+3               |
| Mean cost                    | 19680   | 11114           | 8359                        |

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

Control set  $\mathbb{U}(x)$ : possible points to restart from after an intervention from state  $x$ . For the numerical computation, must be

- ▶ finite
- ▶ the same at any point

For the equipment model, the control set is

- ▶ infinite
- ▶ point dependent as some actions are forbidden in some modes

### Solution

- ▶ discretize the control set
- ▶ manage the point dependency with infinite costs

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

Control set  $\mathbb{U}(x)$ : possible points to restart from after an intervention from state  $x$ . For the numerical computation, must be

- ▶ finite
- ▶ the same at any point

For the equipment model, the control set is

- ▶ infinite
- ▶ point dependent as some actions are forbidden in some modes

### Solution

- ▶ discretize the control set
- ▶ manage the point dependency with infinite costs

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

Control set  $\mathbb{U}(x)$ : possible points to restart from after an intervention from state  $x$ . For the numerical computation, must be

- ▶ finite
- ▶ the same at any point

For the equipment model, the control set is

- ▶ infinite
- ▶ point dependent as some actions are forbidden in some modes

### Solution

- ▶ discretize the control set
- ▶ manage the point dependency with infinite costs

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

Finite control set  $\mathbb{U}$

$\implies$  discretize the functioning times at interventions

$\implies$  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

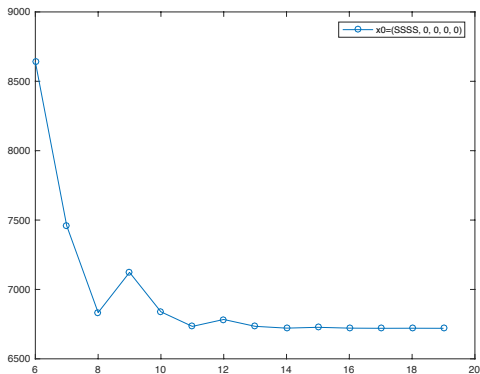
Compromise between precision and computation time

| Number of points | Strategy 1 | Strategy 3 | Strategy 5 |
|------------------|------------|------------|------------|
| 50               | 19680      | 11075      | 8326       |
| 100              | 19680      | 11134      | 8367       |
| 200              | 19680      | 11104      | 8361       |
| 400              | 19680      | 11124      | 8366       |
| 1000             | 19680      | 11109      | 8355       |
| Exact cost       | 19680      | 11114      | 8359       |

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

Horizon  $N$  (number of iterations)

- ▶ 5 for Strategy 1
- ▶ up to 30 for Strategy 3 (mean 6)
- ▶ up to 25 for Strategy 5 (mean 6)



## Step 5: Approximation of the value function

| Strategy<br>1 | Strategy<br>3 | Strategy<br>5 | Approx.<br>Value function |
|---------------|---------------|---------------|---------------------------|
| 19680         | 11114         | 8359          | 6720                      |

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

## Step 6: Optimally controlled trajectories

| Strategy<br>1 | Strategy<br>3 | Strategy<br>5 | Approx.<br>Value function | Optimally<br>controlled traj. |
|---------------|---------------|---------------|---------------------------|-------------------------------|
| 19680         | 11114         | 8359          | 6720                      | 6735                          |

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

# Outline

Introduction

Impulse control for PDMPs

Numerical implementation

Conclusion and perspectives

# Conclusion

## Numerical method to approximate the value function

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

## Work in progress

- ▶ validation of the optimal strategy
- ▶ reference strategy 6 ?
- ▶ sensitivity analysis

# Conclusion

## Numerical method to approximate the value function

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

## Work in progress

- ▶ validation of the optimal strategy
- ▶ reference strategy 6 ?
- ▶ sensitivity analysis

## References

- [CD 89] O. COSTA, M. DAVIS *Impulse control of piecewise-deterministic processes*
- [Davis 93] M. DAVIS, *Markov models and optimization*
- [dSD 12] B. DE SAPORTA, F. DUFOUR *Numerical method for impulse control of piecewise deterministic Markov processes*
- [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. PRINTEMPS *An optimal Markovian quantization algorithm for multi-dimensional stochastic control problems*