



**HAL**  
open science

# Numerical Method for Control of Piecewise-Deterministic Markov Processes

Benoîte de Saporta, François Dufour

► **To cite this version:**

Benoîte de Saporta, François Dufour. Numerical Method for Control of Piecewise-Deterministic Markov Processes. Romain Azaïs; Florian Bouguet. Statistical Inference for Piecewise-deterministic Markov Processes, Wiley, chapter 5, 2018, 9781119544036. hal-01865551

**HAL Id: hal-01865551**

**<https://hal.science/hal-01865551>**

Submitted on 6 Sep 2018

**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.

---

## Numerical Method for Control of Piecewise-Deterministic Markov Processes

---

The aim of this chapter is to present the computational method developed in [DES 10] for the control of a Piecewise-Deterministic Markov Process (PDMP)  $(X_t)_{t \geq 0}$  and to explain how similar ideas can be used for statistical inference. Most technical details are omitted to focus on the practical application of the procedure: we state in details the class of PDMPs it can be applied to and give all the algorithms necessary to its implementation. The main material of this chapter was originally published as [DES 10] and [BRA 12b]. Additional details and examples can also be found in [DES 15].

### 6.1. Introduction

Roughly speaking, a control problem is an optimization problem where a controller acts on a process, either continuously on the flow  $\Phi$  or jump rate  $\lambda$  or punctually at jump times on the jump kernel  $Q$  or even by creating new jumps and selecting new post-jump locations, in order to maximize a gain or minimize a cost. In this chapter, we deal with the most simple form of control, namely optimal stopping. The controller only chooses an admissible time to stop the process in order to maximize a reward and the dynamics of the process before stopping is not controlled.

We are interested in the optimal stopping problem with random horizon  $T_N$ , where  $N \geq 0$  is a nonnegative integer and  $T_N$  the  $N$ -th jumps time of the PDMP. The family  $\mathcal{M}_N$  of admissible times are the stopping times with respect to the natural filtration  $(\mathcal{F}_t)_{t \geq 0}$  of the PDMP  $(X_t)_{t \geq 0}$  that are bounded by  $T_N$ . The reward is the expectation

of some function  $g$  of the process at the chosen stopping time. The *value function*  $v$  is the best possible performance. It depends on the deterministic starting point  $x_0$  of the process

$$v(x_0) = \sup_{\tau \in \mathcal{M}_N} \mathbb{E}_{x_0}[g(X_\tau)]. \quad [6.1]$$

An *optimal stopping time* is a stopping time  $\tau^* \in \mathcal{M}_N$  reaching this supremum:  $v(x_0) = \mathbb{E}_{x_0}[g(X_{\tau^*})]$ . In general, optimal stopping times do not exist and the next best thing is a so-called  $\epsilon$ -*optimal stopping time*  $\tau_\epsilon^* \in \mathcal{M}_N$  satisfying

$$v(x_0) - \epsilon \leq \mathbb{E}_{x_0}[g(X_{\tau_\epsilon^*})] \leq v(x_0).$$

Our objective is to propose a numerically tractable approximation of the value function  $v$ , and to construct a simulatable  $\epsilon$ -optimal stopping time.

Optimal stopping problems for PDMPs have been extensively studied in the literature from the theoretical point of view, see e.g. [COS 88, COS 00, DAV 93, GAT 91, GUG 86, LEN 85]. In [GUG 86] the author defines a dynamic programming operator related to the first jump time of the process, and shows that the value function of the optimal stopping problem is a fixed point for this operator. The basic assumption in this case is that the final cost function is continuous along trajectories, and it is shown that the value function will also have the same property. In [GAT 91, LEN 85] the authors adopt some stronger continuity assumptions and boundary conditions to show that the value function of the optimal stopping problem satisfies some variational inequalities, related to integro-differential equations. In [DAV 93], M.H.A. Davis assumes that the value function is bounded and locally Lipschitz along trajectories to show that the variational inequalities are necessary and sufficient to characterize the value function of the optimal stopping problem. In [COS 00], the authors weakened the continuity assumptions of [DAV 93, GAT 91, LEN 85]. To the best of our knowledge, the only paper addressing computational issues for optimal stopping of PDMPs is [COS 88]. The numerical approach uses a fixed discretization of the state space based on the kernel  $Q$ . The authors derive a convergence result for the approximation scheme but no estimation of the rate of convergence is provided. They also provide  $\epsilon$ -optimal stopping times based on the level sets of the approximate value function, but they require a continuous-time minimization and are not fully numerically tractable, except for simple special cases.

Although the literature on numerical methods for PDMPs is still quite scarce, that for diffusion processes is especially rich. We focused our attention on the optimal quantization approach developed for numerical probability in the series of papers [PAG 98, BAL 03, PAG 04b, PAG 04a, BAL 05]. Optimal quantization consists in replacing a continuous state-space random variable by the best discrete one in the

sense that the  $L_2$ -norm of the difference is minimum. The difference is given by Zador theorem [ZAD 82] and goes to zero as the number of points  $K$  in the support of the discrete distribution goes to infinity, at a speed proportional to  $K^{-1/d}$  where  $d$  is the dimension of the original random variable. In reasonably small dimension, this result is especially appealing as it can lead to explicit error bounds for approximation schemes based on quantization. In addition, there exist simulation-based algorithms providing the discrete quantized variable.

Our approach is the following. We start from the dynamic programming operator defined by [GUG 86], and rewrite it as an expectation of a functional of the PDMP. Then, we use the key property that all the randomness of the PDMP is contained in its pots-jump locations and inter-jump times Markov chain  $(Z_n, S_n)_{n \geq 0}$ , see Subsection 1.2.1 in Chapter 1. The dynamic programming operator can thus be rewritten as a conditional expectation involving this Markov chain. The next step is to replace the continuous state space chain  $(Z_n, S_n)_{n \geq 0}$  by its quantized approximation, thus turning conditional expectations into finite weighted sums. The discretized operator can now be iterated numerically, and we estimate the difference between the original value function and this approximation. A by-product of the discretization is a numerically tractable stopping rule that is an  $\epsilon$ -optimal stopping time for the original PDMP.

The main idea of this approach is first to rewrite the quantity to be computed as an expression of the embedded Markov chain  $(Z_n, S_n)_{n \geq 0}$  and second to replace it by its quantized approximation. It is our belief that many problems related to PDMPs, including statistical estimation problems, can be numerically approximated this way. In particular, this idea was successfully used in other control problems, such as optimal stopping problems under partial observation [BRA 13], impulse control problems [DES 12a] or more recently change-point detection problems [CLE 17], but also to approximate expectations of functionals of PDMPs [BRA 12a] or the distribution of exit times [BRA 12b]. We detail the latter problem here to emphasize the versatility of the approach that we propose.

This chapter is organized as follows. As our approach is simulation based, we start by recalling how to simulate a PDMP in Section 6.2. In Section 6.3, we build the numerical approximation of our optimal stopping problem, as well as the proposed  $\epsilon$ -optimal stopping time. In Section 6.4, we explain how similar ideas can be used to estimate the distribution of an exit time. Finally, in Section 6.5, both problems are illustrated on an industrial example of corrosion provided by Astrium.

## 6.2. Simulation of Piecewise-Deterministic Markov Processes

We will assume in the sequel that the flow can be computed analytically, otherwise, one can use any suitably precise differential equation solver. Still, it is

worth noting that very few results in the literature study the additional error caused by the use of a numerical differential equation solver into the global simulation. The only source of randomness in a PDMP comes from the jumps. In order to simulate a PDMP, one must thus be able to simulate its jump times and its post-jump locations. As in most examples simulating post-jump locations is straightforward, we concentrate on simulating the inter jump-times, which boils down to simulating a random variable with time-dependent intensity, as seen from Eq. [1.2] in Chapter 1.

As always, if the cumulative distribution function of the inter-jump time can be inverted analytically, it is preferable to simulate the inter-jump time by direct inversion. This can be done for instance when the intensity is fixed (exponential distribution), or for the Weibull distribution.

If the cumulative distribution function cannot be inverted analytically, one can use the thinning algorithm 6.1 based on the Poisson distribution that simulates a random variable with time dependent intensity. It is especially suitable when one just wants to simulate the embedded Markov chain  $(Z_n, S_n)_{n \geq 0}$  instead of the whole continuous-time trajectory of the PDMP. It only requires a local maximum of the intensity. If the flow must be solved numerically, Gillespie-type algorithms are usually more efficient, see e.g. [LEM 17].

### 6.3. Optimal Stopping

We start with stating the main regularity assumptions that we require in Subsection 6.3.1, then in Subsection 6.3.2 we give the exact form of the dynamic programming operators. In Subsection 6.3.3 we introduce the quantized approximation of the value function as well as an  $\epsilon$ -optimal stopping time.

#### 6.3.1. Assumptions and Notation

In order to fully use the properties of optimal quantization, we make the main assumption that the driving parameters of the PDMP as well as the reward function of the optimal stopping problem are bounded and Lipschitz continuous.

Let  $\mathcal{L}$  be the subset of  $\mathcal{C}_b$  of Lipschitz-continuous functions, and for any function  $w \in \mathcal{L}$ , denote by  $C_w$  its upper bound and by  $L_w$  its Lipschitz constant

$$C_w = \sup_{x \in \mathcal{X}} |w(x)|, \quad L_w = \sup_{x \neq x' \in \mathcal{X}} \frac{|w(x) - w(x')|}{|x - x'|}.$$

Denote by  $\mathcal{L}_\Phi$  the set of functions that are Lipschitz continuous along the flow i.e. the real-valued, bounded, measurable functions  $w$  defined on  $\mathcal{X}$  and satisfying the following conditions:

**Algorithm 6.1:** Thinning algorithm to simulate a time dependent intensity

---

**input** : Intensity function  $\lambda(t)$ , positive real number  $A$   
**output**: One realization of the random variable with intensity  $\lambda(t)$

```

1 begin
2    $a \leftarrow 0$ 
3    $S \leftarrow \emptyset$ 
4   while  $S \neq \emptyset$  do
5     select  $L \geq \sup_{a \leq t \leq a+A} \lambda(t)$ 
6     draw  $n$  from a Poisson distribution of parameter  $L \times A$ 
7     if  $n \neq 0$  then
8       for  $1 \leq k \leq n$  do
9         draw  $u_k$  from a uniform distribution on  $[a; a + A]$ 
10        draw  $v_k$  from a uniform distribution on  $[0; L]$ 
11        if  $v_k \leq \lambda(u_k)$  then
12           $S = S \cup \{u_k\}$ 
13        end
14      end
15    end
16     $a \leftarrow a + A$ 
17  end
18 return:  $\min S$ 

```

---

– for all  $x \in \mathcal{X}$ , the map  $w(\Phi(x, \cdot)) : [0, t^+(x)) \rightarrow \mathbb{R}$  is continuous, the limit  $\lim_{t \rightarrow t^+(x)} w(\Phi(x, t))$  exists and is denoted by  $w(\Phi(x, t^+(x)))$ ,

– there exists  $[w]_1 \in \mathbb{R}_+$  such that for  $x, y \in \mathcal{X}$  and  $t \in [0, t^+(x) \wedge t^+(y)]$ , one has

$$|w(\Phi(x, t)) - w(\Phi(y, t))| \leq [w]_1 |x - y|;$$

– there exists  $[w]_2 \in \mathbb{R}_+$  such that for all  $x \in \mathcal{X}$  and  $t, t' \in [0, t^+(x)]$ , one has

$$|w(\Phi(x, t)) - w(\Phi(x, t'))| \leq [w]_2 |t - t'|;$$

– there exists  $[w]_* \in \mathbb{R}_+$  such that for all  $x, y \in E$ , one has

$$|w(\Phi(x, t^+(x))) - w(\Phi(y, t^+(y)))| \leq [w]_*^E |x - y|.$$

Note that any function  $w$  in  $\mathcal{L}_\Phi$  is also in  $\mathcal{L}$  with  $L_w \leq [w]_1$  as for all  $x \in \mathcal{X}$ ,  $x = \Phi(x, 0)$ . Conversely, if the flow  $\Phi$  and the exit time  $t^+$  are bounded and Lipschitz

continuous:  $\Phi \in \mathcal{L}(\mathcal{X} \times \mathbb{R}_+)$  and  $t^+ \in \mathcal{L}$  then any function  $w \in \mathcal{L}$  is in  $\mathcal{L}_\Phi$  with  $[w]_1 \leq L_w L_\Phi$ ,  $[w]_2 \leq L_w L_\Phi$  and  $[w]_* \leq L_w L_\Phi (L_{t^+} + 1)$ .

We can now state the main assumptions that will be in force throughout this chapter.

ASSUMPTION 6.1.– Assume that:

– the deterministic time to reach the boundary of the state space is bounded and Lipschitz-continuous:  $t^+$  is in  $\mathcal{L}$ ;

– the jump intensity  $\lambda$  is bounded by  $C_\lambda$  and there exists  $[\lambda]_1 \in \mathbb{R}_+$  such that for all  $x, y \in \mathcal{X}$  and  $t \in [0, t^+(x) \wedge t^+(y)]$ , one has

$$|\lambda(\Phi(x, t)) - \lambda(\Phi(y, t))| \leq [\lambda]_1 |x - y|;$$

– the Markov kernel  $Q$  is Lipschitz in the following sense: there exists  $[Q] \in \mathbb{R}_+$  such that for all functions  $w \in \mathcal{L}_\Phi$ ,

1) for all  $x, y \in \mathcal{X}$  and  $t \in [0, t^+(x) \wedge t^+(y)]$ , one has

$$|Qw(\Phi(x, t)) - Qw(\Phi(y, t))| \leq [Q][w]_1 |x - y|,$$

2) for all  $x, y \in \mathcal{X}$ , one has

$$|Qw(\Phi(x, t^+(x))) - Qw(\Phi(y, t^+(y)))| \leq [Q][w]_* |x - y|;$$

– the reward function  $g$  is in  $\mathcal{L}_\Phi$ .

In most practical applications, the physical properties of the system ensure that either  $t^+$  is bounded, or the problem has a natural deterministic time horizon  $T$ . In the latter case, there is no loss of generality in considering that  $t^+$  is bounded by this deterministic time horizon. The other assumptions are technical and easy to check on most applications.

We finish this section with some generic notation. For  $a, b \in \mathbb{R}$ ,  $a \vee b = \max(a, b)$  is maximum of  $a$  and  $b$ . For a Markov kernel  $P$  on  $(\mathcal{X}, \mathcal{B}(\mathcal{X}))$  and functions  $w$  and  $w'$  in  $\mathcal{C}_b$ , set

$$Pw(x) = \int_{\mathcal{X}} w(y)P(x, dy), \quad (wPw')(x) = w(x) \int_{\mathcal{X}} w'(y)P(x, dy),$$

for any  $x \in \mathcal{X}$ .

### 6.3.2. Dynamic Programming

We now recall the dynamic programming equations derived in [GUG 86]. They were originally given in an analytical form depending on the local characteristics  $\Phi$ ,  $\lambda$  and  $Q$  of the PDMP, but we give their form as expectations as it is the suitable form for our intended discretization. For all  $x \in \mathcal{X}$ ,  $w \in \mathcal{L}_\Phi$  and  $t \geq 0$ , one defines

$$\begin{aligned} Hw(x, t) &= w(\Phi(x, t \wedge t^+(x))) \mathbb{P}(S_{n+1} \geq t \wedge t^+(Z_n) | Z_n = x), \\ Iw(x, t) &= \mathbb{E}[w(Z_{n+1}) \mathbf{1}_{\{S_{n+1} < t \wedge t^+(Z_n)\}} | Z_n = x], \\ Jw(x, t) &= Hg(x, t) + Iw(x, t), \\ Kw(x) &= \mathbb{E}[w(Z_{n+1}) | Z_n = x], \\ Lw(x) &= \sup_{u \leq t^+(x)} Jw(x, u) \vee Kw(x). \end{aligned}$$

As  $(Z_n, S_n)_{n \geq 0}$  is a time-homogeneous Markov chain, the quantities above do not depend on  $n$ . Roughly speaking, operator  $L$  selects the best compromise between waiting for the next jump (operator  $K$ ) or stopping at the best position along the flow ( $\sup J$ ). The value function  $v$  can be recursively constructed by iterating operator  $L$  as follows. Set, for  $0 \leq n \leq N$ ,

$$v_N = g, \quad v_n = L(v_{n+1}, g).$$

The last term  $v_0$  of this recurrence is then exactly the value function  $v$ . In order to obtain a numerical approximation of  $v$ , it is thus sufficient to be able to discretize and iterate operator  $L$ . This is not straightforward as  $L$  involves a continuous supremum as well as conditional expectations. Our first step is to replace the recursion on functions  $v_n$  by a recursion on random variables. Set  $V_N = v_n(Z_n) = g(Z_n)$  and, for  $0 \leq n \leq N$ ,

$$\begin{aligned} V_n = v_n(Z_n) &= \sup_{u \leq t^+(Z_n)} \left\{ \mathbb{E}[V_{n+1} \mathbf{1}_{\{S_{n+1} < u\}} + g(\Phi(Z_n, u)) \mathbf{1}_{\{S_{n+1} \geq u\}} | Z_n] \right\} \\ &\vee \mathbb{E}[V_{n+1} | Z_n]. \end{aligned}$$

It is a highly remarkable property of this problem that the dynamic programming equation involving a sequence of functions  $(v_n)$  can simply be rewritten as a self-contained dynamic programming equation involving only the random variables  $(V_n)$ . As random variables are much more tractable numerically than functions, this is a crucial step towards a numerical approximation of the value function.

### 6.3.3. Quantized Approximation

The next step is to replace the Markov chain  $(Z_n, S_n)_{n \geq 0}$  by its optimal quadratic quantization. This can be done for instance by using the Competitive



Learning Vector Quantization (CLVQ) algorithm described in Algorithm 6.2, see e.g. [BAL 03, Section 3.3] for details. Let  $(\Gamma_n)_{0 \leq n \leq N}$  be the optimal quantization grids for  $(Z_n, S_n)_{0 \leq n \leq N}$ , and  $p_{\Gamma_n}$  the nearest-neighbor projection from  $\mathcal{X} \times \mathcal{S}$  onto  $\Gamma_n$ . One thus defines the quantized approximation  $(\hat{Z}_n, \hat{S}_n)_{0 \leq n \leq N}$  of  $(Z_n, S_n)_{0 \leq n \leq N}$  by

$$(\hat{Z}_n, \hat{S}_n) = p_{\Gamma_n}(Z_n, S_n).$$

We also denote by  $\Gamma_n^Z$  the projection of  $\Gamma_n$  onto  $\mathcal{X}$ .

For  $z \in \mathcal{X}$ , set  $\Delta(z) \in (0, t^+(z))$ . Define  $n(z) = \lfloor \frac{t^+(z)}{\Delta(z)} \rfloor - 1$ , where  $\lfloor x \rfloor$  denotes the greatest integer smaller than or equal to  $x$ . The set of points  $(t_i)_{0 \leq i \leq n(z)}$  with  $t_i = i\Delta(z)$  is denoted by  $G(z)$ . This is the grid associated to the time interval  $[0, t^+(z)]$ .

The process  $(\hat{Z}_n, \hat{S}_n)_{0 \leq n \leq N}$  is not a Markov chain and especially not time-homogeneous. Therefore we define a sequence of approximate operators. For  $n \in \{1, \dots, N\}$ ,  $w \in \mathcal{L}(\Gamma_n^Z)$ ,  $z \in \Gamma_{n-1}^Z$ , and  $s \in \mathbb{R}_+$ , set

$$\begin{aligned} \hat{J}_n(w, g)(z, s) &= \mathbb{E}[w(\hat{Z}_n) \mathbf{1}_{\{\hat{S}_n < s\}} + g(\Phi(z, s)) \mathbf{1}_{\{\hat{S}_n \geq s\}} | \hat{Z}_{n-1} = z], \\ \hat{K}_n(w)(z) &= \mathbb{E}[w(\hat{Z}_n) | \hat{Z}_{n-1} = z], \\ \hat{L}_n^d(w, g)(z) &= \max_{s \in G(z)} \{ \hat{J}_n(w, g)(z, s) \} \vee \hat{K}_n(w)(z). \end{aligned}$$

From these operators, we define a new sequence of random variables  $\hat{V}_n = \hat{v}_n(\hat{Z}_n)$  where  $\hat{v}_N(z) = g(z)$  for  $z \in \Gamma_N^Z$  and

$$\hat{v}_{n-1}(z) = \hat{L}_n^d(\hat{v}_n, g)(z), \quad [6.2]$$

for  $1 \leq n \leq N$  and  $z \in \Gamma_{n-1}^Z$ . Note that the random variables  $\hat{V}_n$  are discrete and can be recursively computed as finite weighted sums. The last term of the recurrence  $\hat{V}_0$  is our chosen approximation for the value function  $v$  at the starting point  $x_0$  of the PDMP. It is detailed in Algorithm 6.3. It also contains additional information related to which operator won the maximization at each step that will be useful to construct an  $\epsilon$ -optimal stopping time.

Under our regularity assumptions, we have the following convergence result. Its proof can be found in [DES 10] or [DES 15, Chapter 7].

**THEOREM 6.1.**– The difference  $\|\hat{V}_0 - V_0\|$  goes to zero as the number of points in the quantization and time discretization grids go to infinity.

**Algorithm 6.2:** CLVQ algorithm to quantize a Markov chain

---

**input** : Number of points  $K$  in each grid, Number of runs  $NR$ , Horizon  $N$ ,  
Initial grids  $(\Gamma_n^0)_{0 \leq n \leq N}$  with  $K$  points each, Sequence  $(\gamma_n)$   
Simulator of trajectories of target Markov chain

**output**: Optimized grids  $(\Gamma_n^*)$  and weights  $(W_n)_{0 \leq n \leq N}$ , Transition probabilities  $P_n, 1 \leq n \leq N$

```

1 begin
2   w ← 0 (size  $K \times N + 1$ )
3   p ← 0 (size  $K \times K \times N$ )
4   for m ← 0 to NR - 1 do
5     simulate trajectory  $(x_0, x_1, \dots, x_N)$  according to law of Markov chain
6     for n ← 0 to N do
7       competitive phase: select  $i_n$  in  $\{1, \dots, K\}$  such that  $y_{i_n}^m$  is the
8         closest neighbor of  $x_n$  in  $\Gamma_n^m$ 
9       learning phase:  $y_{i_n}^{m+1} \leftarrow y_{i_n}^m - \gamma_{n+1}(y_{i_n}^m - x_n)$ 
10      for  $j \neq i_n$  do
11        |  $y_j^{m+1} \leftarrow y_j^m$ 
12      end
13       $\Gamma_n^{m+1} \leftarrow \{y_i^{m+1}, 1 \leq i \leq K\}$ 
14      updating weights:  $w_{i_n, n} \leftarrow w_{i_n, n} + 1$ 
15      if  $n > 0$  then
16        |  $p_{i_{n-1}, i_n, n} \leftarrow p_{i_{n-1}, i_n, n} + 1$ 
17      end
18    end
19    normalizing weights
20    for  $0 \leq n \leq N$  do
21      for  $1 \leq i \leq K$  do
22        |  $(W_n)_i \leftarrow w_{i, n} / NR$ 
23      end
24    end
25    for  $1 \leq n \leq N$  do
26      for  $1 \leq i \leq K$  do
27        for  $1 \leq j \leq K$  do
28          |  $(P_n)_{i, j} \leftarrow p_{i, j, n} / NR$ 
29        end
30      end
31    end
32  return:  $(\Gamma_n^{NR}), W_n, 0 \leq n \leq N, P_n, 1 \leq i \leq N$ .

```

---

**Algorithm 6.3:** Approximation of the value function  $v(x_0)$ 


---

**input** : Quantization grids  $(\Gamma_n^\Theta)_{0 \leq n \leq N}$ , Transition matrices  $(P_n)_{1 \leq n \leq N}$ , cost function  $g$ , time discretization step  $\Delta$

**output** :  $\hat{v}_n$ , indices for  $\epsilon$ -optimal stopping time:  $s_n^*, i_n, 1 \leq n \leq N$

```

1 begin
2   for  $z \in \Gamma_N^Z$  do
3      $\hat{v}_N(z) \leftarrow g(z)$ 
4   end
5   for  $n \leftarrow N - 1$  to 0 do
6     for  $z \in \Gamma_n^Z$  do
7        $\hat{K}\hat{v}_{n+1}(z) \leftarrow \sum_{(z', s') \in \Gamma_{n+1}} P_{n+1}(z, (z', s')) v_{n+1}(z')$ 
8       for  $u \leftarrow 0$  to  $t^*(z) - \Delta$  by step:  $\Delta$ 
9         do
10           $\hat{J}\hat{v}_{n+1}(z, u) \leftarrow \sum_{(z', s') \in \Gamma_{n+1}} P_{n+1}(z, (z', s')) (v_{n+1}(z') \mathbb{1}_{\{s' < u\}} +$ 
11             $g(\Phi(z, u)) \mathbb{1}_{\{s' \geq u\}})$ 
12          end
13           $\hat{v}_n(z) \leftarrow \max_u \{\hat{J}\hat{v}_{n+1}(z, u)\} \vee \hat{K}\hat{v}_{n+1}(z)$ 
14           $s_{n+1}^*(z) \leftarrow \min \{u^*; \hat{J}\hat{v}_{n+1}(z, u^*) = \max_u \{\hat{J}\hat{v}_{n+1}(z, u)\}\}$ 
15          if  $\hat{K}\hat{v}_{n+1}(z) > \max_u \{\hat{J}\hat{v}_{n+1}(z, u)\}$  then
16             $i_{n+1}(z) \leftarrow 1$ 
17          else
18             $i_{n+1}(z) \leftarrow 0$ 
19          end
20        end
21      end
22    end
23  return :  $\hat{v}_0$ 
24 end

```

---

We can also obtain an upper bound for the error in terms of the regularity constants and the quantization error.

More importantly for applications, from this procedure one can also construct an  $\epsilon$ -optimal stopping time, see Algorithm 6.4. Its rigorous construction and the proof that it is a stopping time for the original PDMP and  $\epsilon$ -optimal can be found in [DES 10]. An example of application of this procedure is given in Section 6.5, other examples can be found in [DES 15, Chapter 9].

**Algorithm 6.4:** Computation of an  $\epsilon$ -optimal stopping time

**input** : Quantization grids  $(\Gamma_n)_{0 \leq n \leq N}$ , Outputs of Algorithm 6.3:  $s_n^*$ ,  $i_n$ ,  $1 \leq n \leq N$ , positive real number  $\beta$ , simulator of trajectories of the PDMP, projection operators  $p_{\Gamma_n}$  onto the quantization grids

**output**: Stopping date  $\tau$

```

1 begin
2    $n \leftarrow 0$ 
3    $\tau \leftarrow \text{NaN}$ 
4   while  $n < N$  do
5     simulate  $(Z_n, S_n, S_{n+1})$ 
6      $(z, s) \leftarrow p_{\Gamma_n}(Z_n, S_n)$ 
7     if  $i_{n+1}(z) = 1$  then
8        $r \leftarrow t^*(Z_n)$ 
9     else
10       $r \leftarrow s_{n+1}(z) \mathbb{1}_{\{s_{n+1}(z) < t^*(Z_n)\}} + (t^*(Z_n) - \beta) \mathbb{1}_{\{s_{n+1}(z) \geq t^*(Z_n)\}}$ 
11    end
12    if  $r < S_{n+1}$  then
13       $\tau \leftarrow T_n + r$ 
14       $n \leftarrow N$ 
15    else
16       $n \leftarrow n + 1$ 
17    end
18  end
19  if  $n = N$  and  $\tau = \text{NaN}$  then
20     $\tau \leftarrow T_N$ 
21  end
22  return:  $\tau$ 
23 end

```

**6.4. Exit Time**

Our approach to approximate the optimal stopping problem can be summarized as follows. First, we obtain a recursive construction of the quantity of interest, namely the dynamic programming equations for the value function. Hopefully, this can generically be done using the Markov property. Second, we rewrite the recursion operator in terms of the embedded Markov chain  $(Z_n, S_n)_{n \geq 0}$ . Again, this can be generically done as this chain is the only source of randomness in a PDMP. Third, we replace the continuous state space chain  $(Z_n, S_n)_{n \geq 0}$  by its quantized approximation  $(\hat{Z}_n, \hat{S}_n)_{n \geq 0}$  leading to finite weighted sums. This way, one obtains a fully computable approximation, and one can hope to obtain error bounds and even convergence rates providing the underlying functional are Lipschitz-continuous. It is

our belief that this program can be successfully applied to other problems related to PDMPs, and in particular to statistical estimation problems.

To illustrate this potential, we present in this section the exit time distribution estimation procedure detailed in [BRA 12b] or [DES 15, Chapter 5]. Our aim is not to provide all the technical details, but only to detail the class of PDMPs it can be applied to and the algorithms. We define the exit time problem in Subsection 6.4.1, then we give the recursive formulation of the problem in Subsection 6.4.2.

#### 6.4.1. Problem Setting and Assumptions

Let  $(X_t)_{t \geq 0} = (Y_t, M_t)_{t \geq 0}$  be a PDMP on the state space  $\mathcal{X} = \{(y, m), m \in \mathcal{M}, y \in \mathcal{Y}_m\}$  with mode set  $\mathcal{M}$ . For all mode  $m \in \mathcal{M}$ , let  $U_m$  be an open subset of  $\mathcal{Y}_m$  and set  $U = \{(y, m), m \in \mathcal{M}, y \in U_m\}$ . We deal with the exit time  $\tau$  of  $(X_t)_{t \geq 0}$  from  $U$

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

Denote by  $\mu$  the distribution of the initial state of the process  $X_0 = Z_0$ . To make our exit problem relevant, we assume that the process starts in  $U$  and eventually leaves it almost surely i.e. the support of  $\mu$  is included in  $U$  and  $\mathbb{P}_\mu(\tau < +\infty) = 1$ . The aim of this chapter is to provide an approximation scheme for the survival function of  $\tau$ .

As our approach is based on a recursive computation using the underlying discrete-time Markov chain of the PDMP, we actually use a computation horizon that equals a jump time of the process: we will study  $\tau \wedge T_N$  rather than  $\tau$  itself. We assume that the PDMP is non explosive, i.e.  $\lim_{n \rightarrow \infty} T_n = +\infty$ , so that as  $N$  goes to infinity, one gets

$$\tau \wedge T_N \rightarrow \tau.$$

We assume that  $N$  is chosen such that  $\mathbb{P}_\mu(\tau > T_N)$  is small enough.

Let us define  $u^*(x)$  for all  $x \in U$  as the time for the flow starting from the point  $x$  to exit from  $U$

$$u^*(x) = \inf\{s \geq 0 : \Phi(x, s) \notin U\},$$

where  $u^*$  will play a similar role as  $t^+$  in the previous section. We make the following assumptions.

ASSUMPTION 6.2.– Assume that:

- the function  $u^*$  is in  $\mathcal{L}$  with upper bound  $C_{u^*}$  and Lipschitz constant  $L_{u^*}$ ;

- for all  $m \in \mathcal{M}$ , the set  $U_m$  is convex;
- for  $\alpha > 0$ , set

$$U^\alpha = \{x \in \mathcal{X} : \min\{|x - y| : y \in \partial U\} \leq \alpha\}.$$

There exist  $C_U > 0$  and  $\beta > 0$  such that for all  $0 \leq n \leq N$ ,  $\mathbb{P}_\mu(Z_n \in U^\alpha) \leq C_U \alpha^\beta$ .

The last two assumptions are technical and ensure that the difference between  $Z_n$  and its quantized approximation  $\widehat{Z}_n$  can be controlled even though they are close to the boundary of  $U$ . Finally, we assume that once the process has exited  $U$ , it cannot get back in  $U$ . If it is not the case naturally, one may just kill the process as it exits  $U$ .

#### 6.4.2. Recursive Formulation

One can obtain a recursive construction of the survival function of the truncated exit time as follows. For all  $s > 0$ , define the sequences  $(p_n(s))_{n \in \mathbb{N}}$ ,  $(q_n)_{n \in \mathbb{N}}$  and  $(r_n(s))_{n \in \mathbb{N}}$  by

$$\begin{aligned} p_n(s) &= \mathbb{P}_\mu(\tau > s | \tau \leq T_n), \\ q_n &= \mathbb{P}_\mu(\tau \leq T_n), \\ r_n(s) &= \mathbb{P}_\mu(\{\tau > s\} \cap \{T_n < \tau \leq T_{n+1}\}). \end{aligned}$$

The conditional probability  $p_n(s)$  does not exist when  $q_n = 0$ . We then choose to extend the sequence by setting  $p_n(s) = 0$  in this case. Our objective is to approximate  $p_N(s)$ . The following proposition provides a recursion for the sequence  $(p_n)_{n \leq N}$ , pointing out that  $p_N$  may be computed as soon as the sequences  $(q_n)_{n \leq N}$  and  $(r_n)_{n \leq N-1}$  are known.

**PROPOSITION 6.1.** – For all  $n \in \mathbb{N}$ ,  $s > 0$ , one has  $p_0(s) = 0$  and

$$p_{n+1}(s) = \frac{p_n(s)q_n + r_n(s)}{q_{n+1}} \mathbb{1}_{\{q_{n+1} \neq 0\}}.$$

*Proof of Proposition 6.1.*— First, recall that  $T_0 = 0$  so that one has  $p_0 = 0$  since the process starts in  $U$ . Then, let  $n \in \mathbb{N}$  such that  $q_{n+1} \neq 0$  and note that  $\{\tau \leq T_{n+1}\} = \{\tau \leq T_n\} \cup \{T_n < \tau \leq T_{n+1}\}$ . Thus, one has

$$\begin{aligned} p_{n+1}(s) &= \frac{\mathbb{P}_\mu(\{\tau > s\} \cap \{\tau \leq T_{n+1}\})}{\mathbb{P}_\mu(\tau \leq T_{n+1})} \\ &= \frac{\mathbb{P}_\mu(\{\tau > s\} \cap \{\tau \leq T_n\}) + \mathbb{P}_\mu(\{\tau > s\} \cap \{T_n < \tau \leq T_{n+1}\})}{q_{n+1}} \\ &= \frac{p_n(s)q_n + r_n(s)}{q_{n+1}}, \end{aligned}$$

showing the result.  $\square$

It remains to express the above quantities in term of the embedded Markov chain. We work with the post-jump locations and jump times chain  $(Z_n, T_n)_{n \in \mathbb{N}}$  rather than the post-jump locations and inter-jump times chain  $(Z_n, S_n)_{n \in \mathbb{N}}$ . They play similar roles as  $S_n = T_n - T_{n-1}$  for all  $n \geq 1$  and our recursions already involve  $(T_n)$ , therefore it is natural to keep using this sequence.

First, note that

$$(T_n < \tau) = (Z_n \in U), \quad (\tau \leq T_n) = (Z_n \notin U),$$

as the process cannot go back to  $U$  once it exited  $U$ . Moreover, on the event  $(Z_n \in U, Z_{n+1} \notin U)$ , one has

$$\tau = (T_n + u^*(Z_n)) \wedge T_{n+1},$$

as  $u^*$  is the deterministic exit time from  $U$ . Thus, one obtains

$$\begin{aligned} q_n &= \mathbb{E}_\mu[\mathbb{1}_{U^c}(Z_n)], \\ r_n(s) &= \mathbb{E}_\mu[\mathbb{1}_{\{(T_n + u^*(Z_n)) \wedge T_{n+1} > s\}} \mathbb{1}_U(Z_n) \mathbb{1}_{U^c}(Z_{n+1})]. \end{aligned}$$

### 6.4.3. Numerical Approximation

We now naturally define the quantized approximations of the previous sequences. For all  $s > 0$ , define  $(\hat{q}_n)_{n \leq N}$  and  $(\hat{r}_n)_{n \leq N-1}$  by

$$\begin{aligned} \hat{q}_n &= \mathbb{E}_\mu[\mathbb{1}_{U^c}(\hat{Z}_n)], \\ \hat{r}_n(s) &= \mathbb{E}_\mu[\mathbb{1}_{\{(\hat{T}_n + u^*(\hat{Z}_n)) \wedge \hat{T}_{n+1} > s\}} \mathbb{1}_U(\hat{Z}_n) \mathbb{1}_{U^c}(\hat{Z}_{n+1})]. \end{aligned}$$

---

**Algorithm 6.5:** Approximation of  $(q_n)$  and  $(r_n(s))$ 


---

**input** : Quantization grids  $\Gamma_n, 0 \leq n \leq N$ ,  
 Weight vectors  $W_n, 0 \leq n \leq N$ , Transition matrices  $P_n, 1 \leq n \leq N$   
**output:**  $(\hat{q}_n)$  and  $(\hat{r}_n(s))$

```

1 begin
2    $\hat{q}_0 \leftarrow 0$ 
3   for  $n \leftarrow 0$  to  $N - 1$  do
4      $\hat{q}_{n+1} \leftarrow \sum_{\substack{(z,t) \in \Gamma_{n+1} \\ z \notin U}} W_{n+1}(z,t)$ 
5      $\hat{r}_n(s) \leftarrow \sum_{\substack{(z,t) \in \Gamma_n \\ z \in U}} \sum_{\substack{(z',t') \in \Gamma_{n+1} \\ z' \notin U}} \mathbb{1}_{\{(t+u^*(z)) \wedge t' > s\}} W_n(z,t) P_n((z,t), (z',t'))$ 
6   end
7   return:  $(\hat{q}_n), (\hat{r}_n(s))$ 
8 end
    
```

---

It is important to note that both sequences  $(\hat{q}_n)$  and  $(\hat{r}_n(s))$  may be computed easily from the output of the quantization algorithm as shown in Algorithm 6.5. The approximation of  $p_n$  is then readily obtained by plug-in. For all  $s > 0$  and for all  $n \leq N - 1$ , let  $\hat{p}_0(s) = 0$  and

$$\hat{p}_{n+1}(s) = \frac{\hat{p}_n(s)\hat{q}_n + \hat{r}_n(s)}{\hat{q}_{n+1}} \mathbb{1}_{\{\hat{q}_{n+1} \neq 0\}}.$$

We can prove that this approximation scheme converges, see [BRA 12b] or [DES 15, Chapter 5] for the proof.

**THEOREM 6.2.** – For all  $n \leq N$  and for almost every  $s > 0$ , one has

$$\hat{p}_n(s) \rightarrow p_n(s),$$

when the quantization error goes to zero.

## 6.5. Numerical Example

We study the evolution of the thickness of an aluminum metallic structure subject to corrosion. This model was provided by Astrium and is fully described in [DES 12b]. It concerns a small homogeneous structure within a strategic ballistic missile. The missile is stored successively in three different environments, the workshop, the submarine in operation and the submarine in dry-dock. Then it goes



back to the workshop and so on. The missile stays in each environment during a random duration with exponential distribution. Its parameter depends on the environment. At the beginning of its service time, the structure is treated against corrosion. The period of effectiveness of this protection is also random, with a Weibull distribution. The thickness loss only begins when this initial protection is gone. The degradation law for the thickness loss then depends on the environment through two parameters, a deterministic transition period and a random corrosion rate uniformly distributed within a given range. Typically, the workshop and dry-dock are the more corrosive environments. The randomness of the corrosion rate accounts for small variations and uncertainties in the corrosiveness of each environment.

### 6.5.1. Piecewise-Deterministic Markov Model

The dynamics of the thickness loss for the structure can be described by a PDMP. The finite set of modes is

$$\mathcal{M} = \{1, 2, 3\} \times \{0, 1\},$$

where the first coordinate corresponds to the environment: 1 for the workshop, 2 for the submarine in operation and 3 for the dry-dock, and the second coordinate states whether the corrosion has started (1) or not (0). The state space in modes  $m \in \{1, 2, 3\} \times \{0\}$  is

$$\mathcal{Y}_m = (0, +\infty) \times \mathbb{R}_+,$$

with boundary  $\partial\mathcal{Y}_m = \{0\} \times \mathbb{R}_+$ , and the Euclidean variables are  $Y_t = (\gamma_t, a_t)$  the remaining time of effectivity of the protection, and the time since the last change of mode. The state space in modes  $m = (i, 1) \in \{1, 2, 3\} \times \{1\}$  is

$$\mathcal{Y}_m = \mathbb{R}_+ \times [\rho_i^-, \rho_i^+] \times \mathbb{R}_+,$$

with empty boundary and the Euclidean variables are  $Y_t = (d_t, \rho_t, a_t)$  the thickness loss, the corrosion rate of the given environment, and time since the last change of mode.

Originally at time 0, the mode is  $M_0 = (1, 0)$  and  $Y_0 = (\gamma_0, 0)$ , where  $\gamma_0$  is drawn according to a Weibull distribution with parameters  $\alpha = 2.5$  and  $\beta = 11800$  hours<sup>-1</sup>. In other words, the process starts in the workshop environment with a random protection that will be effective during  $\gamma_0$  hours. The flow in modes  $m \in \{1, 2, 3\} \times \{0\}$  is

$$\Phi_m((\gamma, a), s) = (\gamma - s, a + s),$$

with  $t^+(m, \gamma, a) = \gamma$ , meaning that the boundary is reached when the protection is gone. In modes  $m = (i, 1) \in \{1, 2, 3\} \times \{1\}$ , the flow is

$$\Phi_m((d, \rho, a), s) = (d + d_i(\rho, a + s) - d_i(\rho, a), \rho, a + s),$$

where

$$d_i(\rho, s) = \rho(s + \eta_i(e^{-s/\eta_i} - 1)),$$

is the dynamics of the thickness loss in environment  $i$ . The parameters are given in Table 6.1. All the parameters come from expert opinion. The boundary is empty in this case, so that  $t^+(m, d, \rho, a) = +\infty$ . We will discuss in the next section how to change the value of  $t^+$  so that our boundedness assumption may hold true.

The jump intensity in environment  $i$  is a constant  $\lambda_i$ , which value is given in Table 6.1. The jump kernel is as follows:

– if  $m = (i, 0)$ ,  $y = (\gamma, a)$  with  $\gamma > 0$ ,

$$Q((y, m), \{(\gamma, 0)\} \times \{m'\}) = \mathbb{1}_{\{(0, m'=(i+1[3])\}},$$

with  $i + 1[3] = i + 1$  if  $i \leq 2$  and  $3 + 1[3] = 1$ ;

– if  $m = (i, 0)$ ,  $y = (0, a)$ , and  $A$  is in  $\mathcal{B}(\mathbb{R}_+)$

$$Q((y, m), \{0\} \times A \times \{0\} \times \{m'\}) = \mathbb{1}_{\{m'=(i,1)\}} \frac{1}{\rho_i^+ - \rho_i^-} \int_{\rho_i^-}^{\rho_i^+} \mathbb{1}_A(s) ds,$$

where the parameters  $\rho_i^\pm$  are given in Table 6.1;

– if  $m = (i, 1)$ ,  $y = (d, \rho, a)$ , and  $A$  is in  $\mathcal{B}(\mathbb{R}_+)$

$$Q((y, m), \{d\} \times A \times \{0\} \times \{(j, 1)\}) = \mathbb{1}_{\{j=i+1[3]\}} \frac{1}{\rho_j^+ - \rho_j^-} \int_{\rho_j^-}^{\rho_j^+} \mathbb{1}_A(s) ds.$$

Figure 6.1 shows some simulated trajectories of the thickness loss along time. The slope changes correspond to changes of environment. The observed dispersion is characteristic of the random nature of the phenomenon, and illustrates the poorness of an average rule to monitor such a process. One needs to make decisions adapted to each specific trajectory. The structure is considered unusable if the loss of thickness reaches 0.2 mm.

Table 6.1: Numerical values of the parameters of the corrosion model.

	environment 1	environment 2	environment 3
$\lambda_i$ ( $\text{h}^{-1}$ )	$(17520)^{-1}$	$(131400)^{-1}$	$(8760)^{-1}$
$\eta_i$ (h)	30000	200000	40000
$\rho_i^-$ ( $\text{mm}\cdot\text{h}^{-1}$ )	$10^{-6}$	$10^{-7}$	$10^{-6}$
$\rho_i^+$ ( $\text{mm}\cdot\text{h}^{-1}$ )	$10^{-5}$	$10^{-6}$	$10^{-5}$

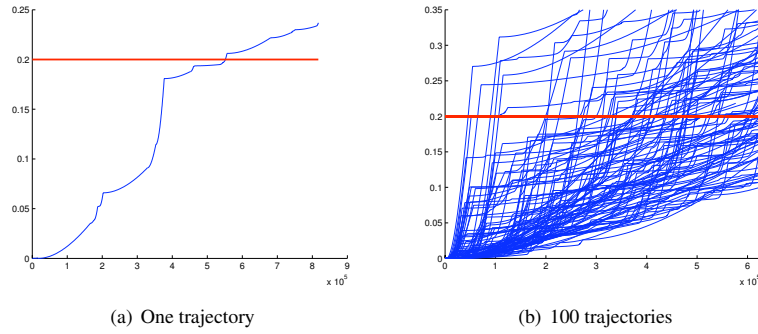


Figure 6.1: Simulated trajectories of the thickness loss along time for the corrosion process.

### 6.5.2. Deterministic Time to Reach the Boundary

Because the deterministic exit time is unbounded in modes  $m \in \{1, 2, 3\} \times \{1\}$ , the PDMP model described above does not fit our framework. There are two main ways to overcome this difficulty.

The first one consists in killing the process when the thickness lost by corrosion reaches the prescribed level of 0.2 mm. The state space in modes  $m = (i, 1) \in \{1, 2, 3\} \times \{1\}$  thus becomes

$$\mathcal{Y}_m = [0, 0.2) \times [\rho_i^-, \rho_i^+] \times \mathbb{R}_+,$$

with boundary  $\partial\mathcal{Y}_m = \{0.2\} \times [\rho_i^-, \rho_i^+] \times \mathbb{R}_+$ , and deterministic time to boundary is

$$t^+((i, 1), d, \rho, a) = \inf\{t > 0 : d + d_i(\rho, a + t) - d_i(\rho, a) = 0.2\}.$$

Thus,  $t^+$  is uniformly bounded on  $\mathcal{Y}_m$  for  $m \in \{1, 2, 3\} \times \{1\}$ . Rigorously, this does not hold true for all points in  $\mathcal{Y}_m$  with  $m \in \{1, 2, 3\} \times \{0\}$  as the support of the

Weibull law is unbounded. However, the initial protection is drawn only once and for a given initial protection time  $\gamma$ , the quantities  $t^+(X_t)$  are bounded uniformly for all  $t$ , which is actually all that is required in our proofs.

The second way to bound the deterministic time to the boundary is by considering the time-augmented process  $(\tilde{X}_t)_{t \geq 0} = (X_t, t)_{t \geq 0}$  with a finite fixed horizon  $T$  for the trajectories. Thus, the state space becomes

$$\mathcal{Y}_m = (0, +\infty) \times \mathbb{R}_+ \times [0, T),$$

in modes  $m \in \{1, 2, 3\} \times \{0\}$ , with boundary

$$\partial\mathcal{Y}_m = \{0\} \times \mathbb{R}_+ \times [0, T) \cup (0, +\infty) \times \mathbb{R}_+ \times \{T\} \cup \{0\} \times \mathbb{R}_+ \times \{T\},$$

and deterministic time to reach the boundary

$$t^+(m, \gamma, a, t) = \gamma \wedge (T - t).$$

The state space in modes  $m = (i, 1) \in \{1, 2, 3\} \times \{1\}$  becomes

$$\mathcal{Y}_m = \mathbb{R}_+ \times [\rho_i^-, \rho_i^+] \times [0, T) \times [0, T),$$

with boundary

$$\partial\mathcal{Y}_m = \mathbb{R}_+ \times [\rho_i^-, \rho_i^+] \times [0, T) \times \{T\},$$

and deterministic time to reach the boundary

$$t^+(m, d, \rho, a, t) = (T - t),$$

as  $a \leq t$  by construction, the boundary may not be reached by  $a$ . Thus,  $t^+$  is bounded by  $T$  uniformly in all the modes and state variables.

### 6.5.3. Quantization

In this example, the main difficulty to implement the CLVQ quantization algorithm lies in the fact that the variables of interest have very different scales: from about  $10^{-6}$  for the corrosion rate  $\rho$  to  $10^5$  for the average time spent in environment 2. This poses a problem as searching for the nearest neighbor and gradient calculations are done in Euclidean norm, regardless of the magnitudes of the components. This defect is corrected by a renormalization of the variables. Therefore a weighted Euclidean norm is used to quantify the Markov chain associated with the corrosion process to

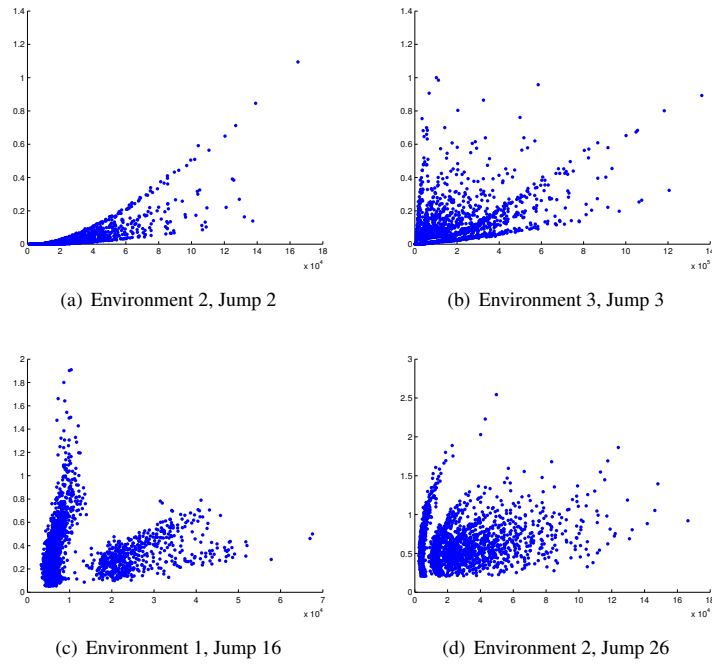


Figure 6.2: Quantization grids (2000 points) for the corrosion process: inter-arrival times (abscissa) and thickness loss at the jump times (ordinate).

rescale the variables. Projections of the quantization grids obtained for 2000 points on the variables inter-arrival times  $S_n$  (abscissa) and thickness loss at the jump times  $d_{T_n}$  (ordinate) are given in Figure 6.2. Notice how different the grids are from one jump time to the other. Note also that the Markov chain does not seem to follow any well-known probability distribution, and that if we had used a cartesian grid on the rectangle, a huge number of points would have been useless. Thus, the quantization approach is especially suitable as it allows for dynamically changing grids along time instead of a single fixed grid.

#### 6.5.4. Optimal Stopping

The reward function  $g$  is given in Figure 6.3. It depends only on the thickness loss, and reflects the fact that on the one hand, one should not perform a maintenance too early, when the thickness loss is low (low gain) but on the other hand one should not cross the security threshold of 0.2 mm (zero gain). It also satisfies our Lipschitz

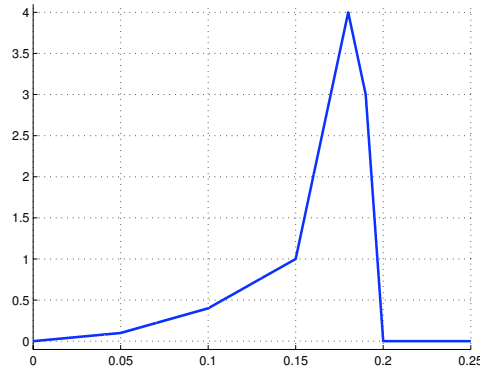


Figure 6.3: Reward function for the maintenance of the corrosion process.

continuity requirements. The reward function  $g$  is maximum (value 4) at abscissa 0.18 and the thickness loss is continuous and non-decreasing. Thus, it is easy to see that the true value function at our starting point is 4, which is the maximum of the reward function  $g$ , and an optimal stopping time is the first moment when the loss reaches 0.18 mm thick.

Our numerical procedure is valid for any sufficiently regular reward function, and the knowledge of the true value function or optimal stopping time shall not be used in the numerical procedure. Besides, recall that the thickness loss is not measured continuously. However, this exact optimal stopping time will serve us as a benchmark to evaluate the performance of our approximation. Figure 6.4 shows two examples of computation of the quasi optimal maintenance time on two specific simulated trajectories. The thick vertical line represents the moment provided by the algorithm to perform maintenance. The other vertical lines materialize the moments of change of environment, the horizontal dotted line the theoretical optimum. In both examples, we stop at a value very close to the optimum value. In addition, the intervention did take place before the critical threshold of 0.2 mm.

An approximate value function  $v$  was calculated in two ways. The first one is the direct method obtained by the algorithm described above. The second one is obtained by Monte Carlo simulation using the quasi-optimal stopping time provided by our procedure. The numerical results obtained are summarized in Table 6.2. As expected, the greater the number of points in the quantization grid, the better our approximation becomes. Furthermore, the specific form of the function  $g$  indicates that at the threshold of 1, the intervention takes place between 0.15 and 0.21 mm and when the threshold increases, this range is narrowed. Our approximation is therefore good even

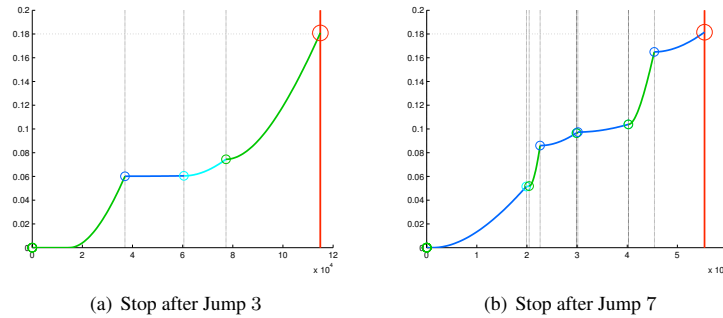


Figure 6.4: Examples of stopped trajectories for the corrosion process.

Table 6.2: Numerical results for the calculation of the value function for the corrosion process.

Number of points in the quantization grids	Approximation of the value function by the direct algorithm	Approximation of the value function by Monte Carlo with the quasi-optimal stopping time
10	2.48	0.94
50	2.70	1.84
100	2.94	2.10
200	3.09	2.63
500	3.39	3.15
1000	3.56	3.43
2000	3.70	3.60
5000	3.82	3.73
8000	3.86	3.75

for low numbers of grid points. The last column of the table also shows the validity of our stopping rule. It should be noted here that this rule does not use the optimal stopping time *stop at the first moment when the thickness loss reaches 0.18 mm*. The method that we propose is general, and implementable even when the optimal stopping time is unknown or does not exist, as will be illustrated in the following sections.

Moreover, one can also construct a histogram of the values of our stopping time, that is to say, a histogram of the values of effective moments of maintenance and compare it with the exact optimal stopping time. Figure 6.5 shows the results for  $10^5$  Monte Carlo simulations. Image (a) is obtained by our approximation procedure and

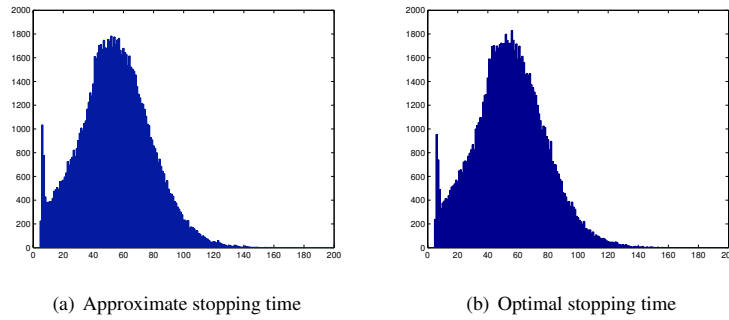


Figure 6.5: Histograms of approximate and optimal stopping times for the corrosion process ( $10^5$  Monte Carlo simulations).

Table 6.3: Approximation results for the distribution of the exit time.

Number of points in the quantization grids	Approximation $\max_s  p_N(s) - \hat{p}_N(s) $
20	0.145
50	0.119
100	0.040
200	0.039
500	0.020

image (b) is the exact one. Both figures appear very similar, which is a strong point in favor of our approximation.

### 6.5.5. Exit Time

Considering the approximation scheme for the exit time distribution, one may note that the quantized value  $\hat{p}_N(s)$  is not necessary smaller than 1. Therefore, it appears natural to replace  $\hat{p}_N(s)$  by  $\hat{p}_N(s) \wedge 1$ . This does not change the convergence theorem and can only improve the approximation error. Figure 6.6 presents the survival function of  $\tau$  obtained through Monte Carlo simulations (the dashed line), through our approximation scheme (the solid line) and the error. Table 6.3 contains the empirical error for different numbers of points in the quantization grids.



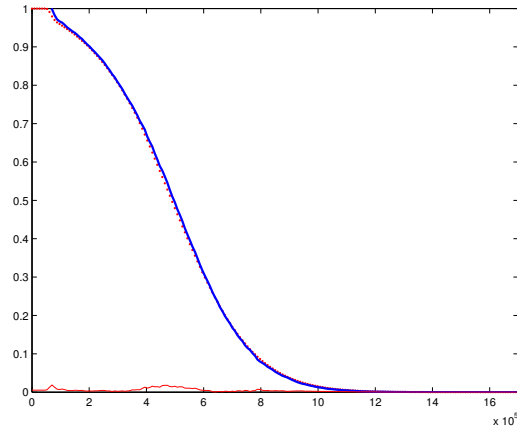


Figure 6.6: Survival function of the service time of the corrosion process obtained through Monte Carlo simulations (dashed), quantized approximation (solid) and the error with 2000 points in the quantization grids.

## 6.6. Conclusion

We have presented a general framework for obtaining numerical approximations of quantities related to PDMPs. First, one obtains a recursive formulation of the quantity to be computed, thanks to the Markov property. Second, one rewrites the expression using the embedded Markov chain  $(Z_n, S_n)$ . Third, one replaces this Markov chain by its quantized approximation in order to obtain fully computable quantities. This approach has been successfully conducted on an optimal stopping problem as well as on an exit time estimation problem.

Computing the quantization grids may be very time consuming, and will lead to poor results in high dimension. However, in small enough dimension, this computation can be done off-line in advance. The same grids can also be reused to compute different quantities. Thankfully, they require only a simulator of the process to be computed. One could also imagine to use observation data to construct such grids, as in [CHA 15], if enough data is available. This approach can also be used in other statistical inference problems.

## 6.7. Bibliography

[BAL 03] BALLY V., PAGÈS G., “A quantization algorithm for solving multi-dimensional discrete-time optimal stopping problems”, *Bernoulli*, vol. 9, num. 6, p. 1003–1049, 2003.

- [BAL 05] BALLY V., PAGÈS G., PRINTEMS J., “A quantization tree method for pricing and hedging multidimensional American options”, *Math. Finance*, vol. 15, num. 1, p. 119–168, 2005.
- [BRA 12a] BRANDEJSKY A., DE SAPORTA B., DUFOUR F., “Numerical method for expectations of piecewise deterministic Markov processes”, *Commun. Appl. Math. Comput. Sci.*, vol. 7, num. 1, p. 63–104, 2012.
- [BRA 12b] BRANDEJSKY A., DE SAPORTA B., DUFOUR F., “Numerical methods for the exit time of a piecewise-deterministic Markov process”, *Adv. in Appl. Probab.*, vol. 44, num. 1, p. 196–225, 2012.
- [BRA 13] BRANDEJSKY A., DE SAPORTA B., DUFOUR F., “Optimal stopping for partially observed piecewise-deterministic Markov processes”, *Stochastic Process. Appl.*, vol. 123, num. 8, p. 3201–3238, 2013.
- [CHA 15] CHARLIER I., PAINDAVEINE D., SARACCO J., “Conditional quantile estimation based on optimal quantization: from theory to practice”, *Comput. Statist. Data Anal.*, vol. 91, p. 20–39, 2015.
- [CLE 17] CLEYNEN A., DE SAPORTA B., Change-point detection for Piecewise Deterministic Markov Processes, ArXiv e-prints, 2017.
- [COS 88] COSTA O. L. V., DAVIS M. H. A., “Approximations for optimal stopping of a piecewise-deterministic process”, *Math. Control Signals Systems*, vol. 1, num. 2, p. 123–146, 1988.
- [COS 00] COSTA O. L. V., RAYMUNDO C. A. B., DUFOUR F., “Optimal stopping with continuous control of piecewise deterministic Markov processes”, *Stochastics Stochastics Rep.*, vol. 70, num. 1-2, p. 41–73, 2000.
- [DAV 93] DAVIS M. H. A., *Markov models and optimization*, vol. 49 of *Monographs on Statistics and Applied Probability*, Chapman & Hall, London, 1993.
- [DES 10] DE SAPORTA B., DUFOUR F., GONZALEZ K., “Numerical method for optimal stopping of piecewise deterministic Markov processes”, *Ann. Appl. Probab.*, vol. 20, num. 5, p. 1607–1637, 2010.
- [DES 12a] DE SAPORTA B., DUFOUR F., “Numerical method for impulse control of piecewise deterministic Markov processes”, *Automatica*, vol. 48, num. 5, p. 779–793, 2012.
- [DES 12b] DE SAPORTA B., DUFOUR F., ZHANG H., ELEGBEDE C., “Optimal stopping for the predictive maintenance of a structure subject to corrosion”, *Proceedings of the Institution of Mechanical Engineers, Part O: Journal of Risk and Reliability*, vol. 226, num. 2, p. 169-181, 2012.
- [DES 15] DE SAPORTA B., DUFOUR F., ZHANG H., *Numerical methods for simulation and optimization of piecewise deterministic Markov processes: application to reliability*, Mathematics and statistics series, Wiley-ISTE, 2015, hal-01249897.
- [GAȚ 91] GAȚAREK D., “On first-order quasi-variational inequalities with integral terms”, *Appl. Math. Optim.*, vol. 24, num. 1, p. 85–98, 1991.
- [GUG 86] GUGERLI U. S., “Optimal stopping of a piecewise-deterministic Markov process”, *Stochastics*, vol. 19, num. 4, p. 221–236, 1986.

- [LEM 17] LEMAIRE V., THIEULLEN M., THOMAS N., Exact simulation of the jump times of a class of Piecewise Deterministic Markov Processes, ArXiv e-prints, 2017.
- [LEN 85] LENHART S., LIAO Y. C., “Integro-differential equations associated with optimal stopping time of a piecewise-deterministic process”, *Stochastics*, vol. 15, num. 3, p. 183–207, 1985.
- [PAG 98] PAGÈS G., “A space quantization method for numerical integration”, *J. Comput. Appl. Math.*, vol. 89, num. 1, p. 1–38, 1998.
- [PAG 04a] PAGÈS G., PHAM H., PRINTEMS J., “An optimal Markovian quantization algorithm for multi-dimensional stochastic control problems”, *Stoch. Dyn.*, vol. 4, num. 4, p. 501–545, 2004.
- [PAG 04b] PAGÈS G., PHAM H., PRINTEMS J., “Optimal quantization methods and applications to numerical problems in finance”, *Handbook of computational and numerical methods in finance*, p. 253–297, Birkhäuser Boston, Boston, MA, 2004.
- [ZAD 82] ZADOR P. L., “Asymptotic quantization error of continuous signals and the quantization dimension”, *IEEE Trans. Inform. Theory*, vol. 28, num. 2, p. 139–149, 1982.