Change-point detection for Piecewise Deterministic Markov Processes

Alice Cleynen, Benoîte de Saporta
Institut Montpelliérain Alexander Grothendieck
CNRS, Univ. Montpellier, France
Outline

Motivation: Stochastic control
  Dynamic optimization
  Examples
  Piecewise deterministic Markov Processes
  Impulse control

Change-point detection problem

Numerical approximation

Simulation study

Conclusion and perspectives
Stochastic control problems

Definition

Dynamic decision making problems
Stochastic control problems

Definition

Dynamic decision making problems

- act on a time-dependent process to change its dynamics
Stochastic control problems

Definition

Dynamic decision making problems

- act on a time-dependent process to change its dynamics
  - continuously: use the accelerator pedal in a car
  - punctually: change gear
Stochastic control problems

Definition

Dynamic decision making problems

- act on a time-dependent process to change its dynamics
  - continuously: use the accelerator pedal in a car
  - punctually: change gear
- in order to fulfill some objective: minimize/maximize some criterion
Stochastic control problems

Definition

Dynamic decision making problems

- act on a time-dependent process to change its dynamics
  - continuously: use the accelerator pedal in a car
  - punctually: change gear
- in order to fulfill some objective: minimize/maximize some criterion
  - drive at the maximum authorized speed
  - minimize fuel consumption
Stochastic control problems

Definition

Dynamic decision making problems under uncertainty

- act on a time-dependent process to change its dynamics
  - continuously: use the accelerator pedal in a car
  - punctually: change gear
- in order to fulfill some objective: minimize/maximize some criterion
  - drive at the maximum authorized speed
  - minimize fuel consumption
- in the presence of randomness
Stochastic control problems

Definition

Dynamic decision making problems under uncertainty

- act on a time-dependent process to change its dynamics
  - continuously: use the accelerator pedal in a car
  - punctually: change gear
- in order to fulfill some objective: minimize/maximize some criterion
  - drive at the maximum authorized speed
  - minimize fuel consumption
- in the presence of randomness
  - other cars
  - unknown route
Stochastic control problems

Questions of interest

Dynamic decision making problems under uncertainty

▶ value function: best mean performance
Stochastic control problems

Questions of interest

Dynamic decision making problems under uncertainty

- **value function**: best mean performance
  - regularity properties: continuity, differentiability, convexity
  - characterization as the unique solution to some explicit equation
Stochastic control problems

Questions of interest

Dynamic decision making problems under uncertainty

- **value function**: best mean performance
  - regularity properties: continuity, differentiability, convexity
  - characterization as the unique solution to some explicit equation
  - numerical approximation
Stochastic control problems

Questions of interest

Dynamic decision making problems under uncertainty

- **value function**: best mean performance
  - regularity properties: continuity, differentiability, convexity
  - characterization as the unique solution to some explicit equation
  - numerical approximation
- (near) optimal strategy
Stochastic control problems

Questions of interest

Dynamic decision making problems under uncertainty

- **value function**: best mean performance
  - regularity properties: continuity, differentiability, convexity
  - characterization as the unique solution to some explicit equation
- **(near) optimal strategy**
  - existence? in which form?
  - properties, sensitivity analysis
Stochastic control problems

Questions of interest

Dynamic decision making problems **under uncertainty**

- **value function**: best mean performance
  - regularity properties: continuity, differentiability, convexity
  - characterization as the unique solution to some explicit equation
  - numerical approximation

- **(near) optimal strategy**
  - existence? in which form?
  - properties, sensitivity analysis
  - numerical approximation
Example 1: Medical treatment optimization

[Pasin 18]

- Population: HIV patients
Example 1: Medical treatment optimization

[Pasin 18]

- Population: HIV patients
- Possible actions: cycles of injections of IL
  - number of injections
  - dose
  - dates of injection
Motivation: Stochastic control

Example 1: Medical treatment optimization

[Pasin 18]

- Population: HIV patients
- Possible actions: cycles of injections of IL
  - number of injections
  - dose
  - dates of injection
- Objective: minimize the time spent with low CD4$^+$ T lymphocytes count
Example 1: Medical treatment optimization

[Pasin 18]

- Population: HIV patients
- Possible actions: cycles of injections of IL
  - number of injections
  - dose
  - dates of injection
- Objective: minimize the time spent with low CD4$^+$ T lymphocytes count
- Sources of randomness
  - random response to injections
  - individual variability between patients
Example 1: Medical treatment optimization

[Passin 18]

Examples of optimally controlled $CD4^+$ T trajectories
Example 2: Maintenance optimization

[Geeraert 17]

- Object of interest: multi-component optronic camera
Example 2: Maintenance optimization

[Geeraert 17]

- **Object of interest:** multi-component optronic camera
- **Possible actions:** maintenance
  - repair or replace
  - which components
  - dates of intervention

Sources of randomness:
- random degradation or failure times for each component
Example 2: Maintenance optimization

[Geeraert 17]

- Object of interest: multi-component optronic camera
- Possible actions: maintenance
  - repair or replace
  - which components
  - dates of intervention
- Objective: minimize the unavailability + maintenance cost
Example 2: Maintenance optimization

[Geeraert 17]

- **Object of interest**: multi-component optronic camera
- **Possible actions**: maintenance
  - repair or replace
  - which components
  - dates of intervention
- **Objective**: minimize the unavailability + maintenance cost
- **Sources of randomness**
  - random degradation or failure times for each component
Example 2: Maintenance optimization

[Geeraert 17]

Reference policy

- send camera to the workshop one day after failure or deterioration
- replace failed components, repair degraded components
Example 2: Maintenance optimization

[Geeraert 17]

Reference policy

- send camera to the workshop one day after failure or deterioration
- replace failed components, repair degraded components

Minimal cost (value function)

- maintenance authorized only after failure or deterioration: 20% lower
- maintenance authorized at all times: 38% lower
Common points

- family of underlying stochastic models **PDMPs**
- type of optimization problem: **impulse control**
**Piecewise deterministic Markov processes**

**Davis (80’s)**

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

Starting point

\[ X_0 = (m, x) \]
Motivation: Stochastic control

Piecewise deterministic Markov Processes

**Piecewise deterministic Markov processes**

**Davis (80’s)**

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 $T_1 = S_1$

\[
X_t = (m, \phi_m(x, t)), \quad P_{(m,x)}(S_1 > t) = e^{-\int_0^t \lambda_m(\phi_m(x,s)) \, ds}
\]
Piecewise deterministic Markov processes

Davis (80’s)

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 (80’s)

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 \( T_2 = T_1 + S_2 \)

\[
X_{T_1 + t} = \left( m_1, \phi_{m_1}(x_{T_1}, t) \right), \quad t < S_2
\]
Motivation: Stochastic control

Piecewise deterministic Markov Processes

Piecewise deterministic Markov processes

**Davis (80’s)**

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

\[ Q_{m_1}(\phi_{m_1}(x_{T_1}, S_2), \cdot) \ldots \]
Applications

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 breakdown
- change treatment before relapse
- ...
Impulse control - State of the art

Lots of works on theoretical problems
Impulse control - State of the art

Lots of works on theoretical problems

Few works on numerical approximations

- [CD 89] Numerical approximation of the value function and $\epsilon$-optimal strategy
  - based on a discretization of the state space and Markov kernel
  - requires solving multiple optimal stopping problems
Impulse control - State of the art

Lots of works on theoretical problems

Few works on numerical approximations

- [CD 89] Numerical approximation of the value function and $\epsilon$-optimal strategy
  - based on a discretization of the state space and Markov kernel
  - requires solving multiple optimal stopping problems

- [dSDZ 14] Numerical approximation of the value function
  - based on a time-dependent discretization of an underlying Markov chain
  - work in progress for $\epsilon$-optimal strategy
Impulse control - State of the art

Lots of works on theoretical problems

Few works on numerical approximations

- **[CD 89]** Numerical approximation of the value function and $\epsilon$-optimal strategy
  - based on a discretization of the state space and Markov kernel
  - requires solving multiple optimal stopping problems

- **[dSDZ 14]** Numerical approximation of the value function
  - based on a time-dependent discretization of an underlying Markov chain
  - work in progress for $\epsilon$-optimal strategy

- **[Pasin 18]** Numerical approximation of the value function and $\epsilon$-optimal strategy
  - based on a discretization of the state space and Markov kernel
  - actions can be taken only at the boundary of the state space
  - heuristics, no mathematical proof
Impulse control - State of the art

Lots of works on theoretical problems

Few works on numerical approximations

- [CD 89] Numerical approximation of the value function and $\epsilon$-optimal strategy
  - based on a discretization of the state space and Markov kernel
  - requires solving multiple optimal stopping problems

- [dSDZ 14] Numerical approximation of the value function
  - based on a time-dependent discretization of an underlying Markov chain
  - work in progress for $\epsilon$-optimal strategy

- [Pasin 18] Numerical approximation of the value function and $\epsilon$-optimal strategy
  - based on a discretization of the state space and Markov kernel
  - actions can be taken only at the boundary of the state space
  - heuristics, no mathematical proof

In all cases, the process is perfectly observed at all times
If the jump times are not observed?

Jump times can be
- date when CD4\(^+\) T count reach 500 threshold
- random failure/deterioration dates

Not observed!
If the jump times are not observed?

Jump times can be

- date when CD4$^+$ T count reach 500 threshold
- random failure/deterioration dates

Not observed!

- [BdSD 12] Optimal stopping
  - jump times observed
  - post-jump locations observed through noise
  Numerical approximation of the value function and $\epsilon$-optimal stopping time

- [BL 17] Continuous control
  - jump times observed
  - post-jump locations observed through noise
  Optimality equation, existence of optimal policies
If the jump times are not observed?

Jump times can be

- date when CD$4^+$ T count reach 500 threshold
- random failure/deterioration dates

Not observed!

- [BdSD 12] Optimal stopping
  - jump times observed
  - post-jump locations observed through noise
  Numerical approximation of the value function and $\epsilon$-optimal stopping time

- [BL 17] Continuous control
  - jump times observed
  - post-jump locations observed through noise
  Optimality equation, existence of optimal policies

No information on the jump times $\Rightarrow$ very difficult problem
Change-point detection

Simplest special case

- only one jump of the mode variable
- discrete noisy observations of the continuous variable on a regular time grid

Optimal stopping = Change-point detection

Aim: numerical approximation to

- detect the change-point at best (not too early/late)
- estimate the new mode after the jump
Typical example

- Population: cancer patients
Typical example

- Population: cancer patients
- Possible actions: change treatment
  - treatment 1
  - treatment 2
  - dates of change
Typical example

- Population: cancer patients
- Possible actions: change treatment
  - treatment 1
  - treatment 2
  - dates of change
- Objective: maximize life time of the patient with minimal secondary effect
Typical example

- Population: cancer patients
- Possible actions: change treatment
  - treatment 1
  - treatment 2
  - dates of change
- Objective: maximize life time of the patient with minimal secondary effect
- Sources of randomness
  - relapse date
  - relapse type
- Observations: cancer cell loads (or proxy) at some regularly spaced measurement times, e.g. every 3 month
Outline

Motivation: Stochastic control

Change-point detection problem
  PDMP model
  State of the art on change-point detection
  MDP model

Numerical approximation

Simulation study

Conclusion and perspectives
Simple PDMP model

- State space $E \times \mathbb{R} = \{0, 1, \ldots, d\} \times \mathbb{R} \times \mathbb{R}$: mode, position, time
- Starting point $X_0 = (0, x, 0)$, flow $\Phi_0$
- Time-dependent Jump intensity $\lambda_0(x, u) = \lambda(u)$
- Jump kernel: position and time continuous, switch to mode $i$ with probability $p_i$
Observations

- Observation times $t_n = \delta n$
- Noisy observations of the positions $Y_n = F(x_{tn}) + \epsilon_n$
Observations

- Observation times \( t_n = \delta n \)
- Noisy observations of the positions \( Y_n = F(x_{t_n}) + \epsilon_n \)
Observations

- Observation times $t_n = \delta n$
- Noisy observations of the positions $Y_n = F(x_{t_n}) + \epsilon_n$
Example: flat/exponential model

- $d = 3$ possible post-jump modes, same probability $p_i = 1/3$, starting from $x_0 = 1$
- $\Phi_0(x, t) = x$, $\Phi_1(x, t) = xe^{0.1t}$, $\Phi_2(x, t) = xe^{0.5t}$, $\Phi_3(x, t) = xe^{1t}$
Segmentation

- data collected until the time horizon
- a posteriori reconstruction of the change point
Segmentation

- data collected until the time horizon
- a posteriori reconstruction of the change point

Irrelevant in our medical context: change must be detected as soon as possible
Moving average

- compute the average of past data over a moving window
- detect rupture when the average exceeds some threshold
Moving average

- compute the average of past data over a moving window
- detect rupture when the average exceeds some threshold

Works well if

- data are centered before the rupture
- data have a positive trend after the rupture
- data have low variance
- small time interval between data
Kalman Filter

- discrete-time linear gaussian model observed through gaussian additive noise
- best mean squares approximation of the hidden variable given the observations
- small time interval between data

![Kalman filter graph](image-url)
State of the art on change point detection

No generic method available if

- long interval between 2 observations
- non gaussian-linear model
- non additive noise
- aim is to detect rupture and new mode after rupture
Partially observed optimal stopping problem

- Finite horizon $\delta N$
Partially observed optimal stopping problem

- Finite horizon $\delta N$
- Admissible stopping times $\tau$: $\mathcal{F}^Y$-measurable
- Admissible decisions $A$: $\{0, 1, \ldots, d\}$ valued, $\mathcal{F}^Y_\tau$-measurable
Partially observed optimal stopping problem

- Finite horizon $\delta N$
- Admissible stopping times $\tau$: $\mathcal{F}^Y$-measurable
- Admissible decisions $A$: $\{0, 1, \ldots, d\}$ valued, $\mathcal{F}_\tau^Y$-measurable
- Cost per stage before stopping
  - $c(0, x, y) = 0$ rightfully not stopped
  - $c(m \neq 0, x, y) = \beta_i \delta$ lateness penalty
- Terminal cost at stopping
  - $C(m, x, y, 0) = c(m, x, y)$ no stopping before the horizon
  - $C(0, x, y, a \neq 0) = \alpha$ early stopping penalty
  - $C(m \neq 0, x, y, a = m) = 0$ good mode selection
  - $C(m \neq 0, x, y, a \neq 0, m) = \gamma$ wrong mode penalty
Partially observed optimal stopping problem

- Finite horizon $\delta N$
- Admissible stopping times $\tau$: $\mathcal{F}^Y$-measurable
- Admissible decisions $A$: $\{0, 1, \ldots, d\}$ valued, $\mathcal{F}_\tau^Y$-measurable
- Cost per stage before stopping
  - $c(0, x, y) = 0$ rightfully not stopped
  - $c(m \neq 0, x, y) = \beta_i \delta$ lateness penalty
- Terminal cost at stopping
  - $C(m, x, y, 0) = c(m, x, y)$ no stopping before the horizon
  - $C(0, x, y, a \neq 0) = \alpha$ early stopping penalty
  - $C(m \neq 0, x, y, a = m) = 0$ good mode selection
  - $C(m \neq 0, x, y, a \neq 0, m) = \gamma$ wrong mode penalty

Cost of admissible strategy $(\tau, A)$

$$J(\tau, A, (m, x, y)) = \mathbb{E}_{(m, x, y)} \left[ \sum_{n=0}^{(\tau-1)\wedge N} c(X_n, Y_n) + C(X_{\tau\wedge N}, Y_{\tau\wedge N}, A) \right]$$
Fully observed optimal stopping problem

- Filter process $\Theta_n(A \times B) = \mathbb{P}_{(0,x,y)}(X_{\delta_n} \in A \times B|\mathcal{F}_n^Y)$
- $(\Theta_n, Y_n)$ time inhomogeneous Markov chain with explicit transition kernels $R'_n$ on $\mathcal{P}(E) \times \mathbb{R}$
- Cost functions $c'(\theta, y) = \int c(m, x, y) d\theta(m, x)$, $C'(\theta, y, a) = \int C(m, x, y, a) d\theta(m, x)$

Minimize over all admissible strategies $(\tau, a)$

$$J'(\tau, A, (\theta, y)) = \mathbb{E}_{(\theta,y)} \left[ \sum_{n=0}^{(\tau-1)^\wedge N} c'(\Theta_n, Y_n) + C'(\Theta_{\tau^\wedge N}, Y_{\tau^\wedge N}, A) \right]$$
Aim

- numerical approximation of the value function
- computable (optimal ?) strategy

Difficulties

- measure-valued filter process: recursive equations but not simulatable
- curse of dimensionality
Outline

Motivation: Stochastic control

Change-point detection problem

Numerical approximation
  Approach
  Optimal quantization
  Convergence results
  Computable strategy

Simulation study

Conclusion and perspectives
Dynamic programming

Value function

\[ V'(\theta, y) = \inf_{(\tau, A)} J'(\tau, A, (\theta, y)) \]

\[ = \inf_{(\tau, A)} \mathbb{E}_{(\theta, y)} \left[ \sum_{n=0}^{(\tau-1)\wedge N} c'(\Theta_n, Y_n) + C'(\Theta_{\tau\wedge N}, Y_{\tau\wedge N}, A) \right] \]

Dynamic programming

\[ v'_N(\theta, y) = \min_{0 \leq a \leq d} C'(\theta, y, a) \]

\[ v'_k(\theta, y) = \min \left\{ \min_{1 \leq a \leq d} C'(\theta, y, a); c'(\theta, y) + R'_k v'_{k+1}(\theta, y) \right\} \]

\[ v'_0 = V' \]
Approach

- Discretize the kernels $R'_k$ to discretize the Dynamic programming operators based on simulation-based discretization grids of the chain $(\Theta_n, Y_n)$. 
Approach

- Discretize the kernels $R'_k$ to discretize the Dynamic programming operators based on simulation-based discretization grids of the chain $(\Theta_n, Y_n)$.

Problems

- $\Theta_n$ is not simulatable

\[
\Theta_{n+1}(A) = \frac{\int_X P_n(HY_{n+1} \mathbb{1}_A)(m, x) d\Theta_n(m, x)}{\int_X P_n(HY_{n+1})(m, x) d\Theta_n(m, x)}
\]

- approximation in 2 steps: approximate simulation of $\Theta_n +$ discretization of the approximation
Discretization

\[ X_t = (m_y, x_t, t) \]
\[ E \times \mathbb{R}, P \]
Discretization

\[ X_t = (m_y, x_t, t) \]
\[ E \times \mathbb{R}, P \]
\[ X_n = (m_{t_n}, x_{t_n}) \]
\[ E, P_n \]

observations
\[ Y_n = F(X_n) + \varepsilon_n \]
\[ (X_n, Y_n) \]
\[ E \times \mathbb{R}, R_n \]
Discretization

\[ X_t = (m_y, x_t, t) \]
\[ E \times \mathbb{R}, P \]

\[ X_n = (m_{t_n}, x_{t_n}) \]
\[ E, P_n \]

observations

\[ Y_n = F(X_n) + \epsilon_n \]

\[ (X_n, Y_n) \]
\[ E \times \mathbb{R}, R_n \]

filtering \[ \psi \]

\[ (\Theta_n, Y_n) \]
\[ \mathcal{P}(E) \times \mathbb{R}, R'_n \]

dynamic programming

\[ \nu'_n(\Theta_n, Y_n) \]
Discretization

\[ X_t = (m_y, x_t, t) \]
\[ E \times \mathbb{R}, P \]

\[ X_n = (m_{tn}, x_{tn}) \]
\[ E, P_n \]

(\( \bar{m}_{tn}, \bar{x}_{tn} \) = \( \bar{X}_n \))

quantization

observations

\[ Y_n = F(X_n) + \varepsilon_n \]

\[ (X_n, Y_n) \]
\[ E \times \mathbb{R}, R_n \]

filtering

\[ \psi \]

\[ (\Theta_n, Y_n) \]
\[ \mathcal{P}(E) \times \mathbb{R}, R'_n \]

dynamic programming

dependent programming

\[ \nu'_n(\Theta_n, Y_n) \]
Discretization

\[ X_t = (m_y, x_t, t) \]
\[ E \times \mathbb{R}, P \]

\[ X_n = (m_{tn}, x_{tn}) \]
\[ E, P_n \]

quantization

observations

\[ Y_n = F(X_n) + \varepsilon_n \]

\[ (X_n, Y_n) \]
\[ E \times \mathbb{R}, R_n \]

filtering \( \Psi \)

\[ (\Theta_n, Y_n) \]
\[ \mathcal{P}(E) \times \mathbb{R}, R'_n \]

dynamic programming

\[ \nu'_n(\Theta_n, Y_n) \]

\[ \bar{\nu}'_n(\bar{\Theta}_n, \bar{Y}_n) \]
Discretization

\[
X_t = (m_y, x_t, t) \\
E \times \mathbb{R}, P \\
\downarrow \\
X_n = (m_{t_n}, x_{t_n}) \\
E, P_n \\
\text{quantization} \\
\downarrow \\
(X_n, Y_n) \\
E \times \mathbb{R}, R_n \\
\text{filtering} \quad \Psi \\
\downarrow \\
(\Theta_n, Y_n) \\
\mathcal{P}(E) \times \mathbb{R}, R'_n \\
\text{dynamic programming} \\
\downarrow \\
\nu'_n(\Theta_n, Y_n) \\
(\hat{\Theta}_n, \hat{Y}_n) \\
\mathcal{P}(\Omega_n) \times \mathbb{Y}, R'_n \\
\text{quantization} \\
\downarrow \\
(\hat{\Theta}_n, \hat{Y}_n) \\
\Gamma_n, \hat{R}'_n
\]
Discretization

\[ X_t = (m_y, x_t, t) \]
\[ E \times \mathbb{R}, P \]
\[ X_n = (m_{t_n}, x_{t_n}) \]
\[ E, P_n \rightarrow \Omega_n, \bar{P}_n \]
\[ (X_n, Y_n) \]
\[ E \times \mathbb{R}, R_n \rightarrow \Omega_n \times \mathbb{Y}, \bar{R}_n \]
\[ (\Theta_n, Y_n) \]
\[ \mathcal{P}(E) \times \mathbb{R}, R'_n \rightarrow \mathcal{P}(\Omega_n) \times \mathbb{Y}, \bar{R}'_n \]
\[ \nu'_n(\Theta_n, Y_n) \rightarrow \bar{\nu}'_n(\bar{\Theta}_n, \bar{Y}_n) \rightarrow \hat{\nu}'_n(\hat{\Theta}_n, \hat{Y}_n) \]
Quantization

[P 98], [PPP 04], [PRS05], ...

Quantization of a random variable $X \in L^2(\mathbb{R}^q)$

Approximate $X$ by $\hat{X}$ taking finitely many values such that $\|X - \hat{X}\|_2$ is minimum

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

Asymptotic properties

If $E[|X|^{2+\eta}] < +\infty$ for some $\eta > 0$ then

$$\lim_{K \to \infty} K^{1/q} \min_{|\Gamma| \leq K} \|X - \hat{X}_\Gamma\|_2 = C$$
Algorithms

There exist algorithms providing

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

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

There exist algorithms providing

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

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

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

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

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

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

**Assets**
- A simulator of the target law is enough to build the grids
- Automatic construction of grids
- Convergence rate for $\mathbb{E}[|f(X) - f(\hat{X})|]$ if $f$ is Lipschitz
- Empirical error measure by Monte Carlo

**Drawbacks**
- Computation time
- Curse of dimension
- Open questions of convergence of the algorithms
Convergence

Technical assumptions

\[
|\nu'_0(\delta_{(0,x_0)}, y_0) - \bar{\nu}'_0(\delta_{(0,x_0)}, y_0)| \leq \sum_{n=0}^{N-1} a_n \mathbb{E}[|\bar{X}_n - X_n|] = O(N^{-1}_\Omega)
\]

\[
|\hat{\nu}'_0(\delta_{(0,x_0)}, y_0) - \bar{\nu}'_0(\delta_{(0,x_0)}, y_0)| \\
\leq \sum_{n=0}^{N} c_n \left( \mathbb{E} \left[ |\hat{Y}_n - \bar{Y}_n| \right] + \mathbb{E} \left[ \|\hat{\Theta}_n - \bar{\Theta}_n\|_{n,1} \right] \right) = O(N^{-1}/N_\Omega)
\]
Candidate computable strategy

Dynamic programming

\[ \hat{v}_N'(\hat{\theta}, \hat{y}) = \min_{0 \leq a \leq d} C'(\hat{\theta}, \hat{y}, a) \]
\[ \hat{v}_k'(\hat{\theta}, \hat{y}) = \min \left\{ \min_{1 \leq a \leq d} C'(\hat{\theta}, \hat{y}, a); c'(\hat{\theta}, \hat{y}) + \hat{R}_k' \hat{v}_{k+1}'(\hat{\theta}, \hat{y}) \right\} \]

Set

\[ r_N(\cdot) = 0, \ a_N(\cdot) = 0 \text{ if } \hat{v}_N'(\text{proj}_{\Gamma_1} \cdot) = C'(\text{proj}_{\Gamma_1} \cdot, 0) \]
\[ r_N(\cdot) = 1, \ a_N(\cdot) = i \text{ if } \hat{v}_N'(\text{proj}_{\Gamma_1} \cdot) = C'(\text{proj}_{\Gamma_1} \cdot, i) \]
Candidate computable strategy

Dynamic programming

- \( \hat{v}'_N(\hat{\theta}, \hat{y}) = \min_{0 \leq a \leq d} C'(\hat{\theta}, \hat{y}, a) \)
- \( \hat{v}'_k(\hat{\theta}, \hat{y}) = \min \left\{ \min_{1 \leq a \leq d} C'(\hat{\theta}, \hat{y}, a); c'(\hat{\theta}, \hat{y}) + \hat{R}'_k \hat{v}'_{k+1}(\hat{\theta}, \hat{y}) \right\} \)

Set

- \( r_N(\cdot) = 0, \ a_N(\cdot) = 0 \) if \( \hat{v}'_N(\text{proj}_{\Gamma_N}(\cdot)) = C'(\text{proj}_{\Gamma_N}(\cdot), 0) \)
- \( r_N(\cdot) = 1, \ a_N(\cdot) = i \) if \( \hat{v}'_N(\text{proj}_{\Gamma_N}(\cdot)) = C'(\text{proj}_{\Gamma_N}(\cdot), i) \)
- \( r_n(\cdot) = 0 \) if \( \hat{v}'_n(\text{proj}_{\Gamma_n}(\cdot)) = \hat{R}'_n \hat{v}'_{n+1}(\text{proj}_{\Gamma_n}(\cdot)) \)
- \( r_n(\cdot) = 1, \ a_n(\cdot) = i \) if \( \hat{v}'_n(\text{proj}_{\Gamma_n}(\cdot)) = C'(\text{proj}_{\Gamma_n}(\cdot), i) \)
Path-adapted computable strategy

\[
\begin{align*}
n & \leftarrow 0 \\
y & \leftarrow y_0 \\
\tilde{\theta} & \leftarrow \delta_{(0,x_0)} \\
r & \leftarrow r_0(\theta, y)
\end{align*}
\]
Path-adapted computable strategy

\[
\begin{align*}
  n &\leftarrow 0 \\
y &\leftarrow y_0 \\
\tilde{\theta} &\leftarrow \delta(0, x_0) \\
r &\leftarrow r_0(\theta, y) \\
r = 1 ?
\end{align*}
\]
Path-adapted computable strategy

\[
\begin{align*}
n &\leftarrow 0 \\
y &\leftarrow y_0 \\
\bar{\theta} &\leftarrow \delta_{(0,x_0)} \\
r &\leftarrow r_0(\theta, y)
\end{align*}
\]

Observation \( y_0 \)

yes

\( r = 1 ? \)

Stop at time \( n \)
Choose decision \( a = a_n(\bar{\theta}, y) \)

no

\[
\begin{align*}
n &\leftarrow n + 1 \\
y &\leftarrow y_n \\
\bar{\theta} &\leftarrow \Psi_{n - 1}(\bar{\theta}, y)
\end{align*}
\]
Path-adapted computable strategy

\[ n \leftarrow 0 \]
\[ y \leftarrow y_0 \]
\[ \bar{\theta} \leftarrow \delta_{(0,x_0)} \]
\[ r \leftarrow r_0(\theta, y) \]

Observation \( y_0 \)

\[ r = 1 ? \]
\[ n = N ? \]

\textit{Stop at time} \( n \)
Choose decision \( a = a_n(\bar{\theta}, y) \)
Path-adapted computable strategy

\begin{align*}
    n &\leftarrow 0 \\
y &\leftarrow y_0 \\
\tilde{\theta} &\leftarrow \delta_{(0,x_0)} \\
r &\leftarrow r_0(\tilde{\theta}, y)
\end{align*}

Observation $y_0$

\begin{align*}
r = 1 \text{ ?} & \\
\text{yes} \quad \text{Stop at time } n \\
\text{Choose decision } a = a_n(\tilde{\theta}, y) \\
\text{no} & \\
n = N \text{ ?} & \\
\text{yes} \quad \text{Choose decision } a = 0
\end{align*}
Path-adapted computable strategy

\[
\begin{align*}
n &\leftarrow 0 \\
y &\leftarrow y_0 \\
\bar{\theta} &\leftarrow \delta_{(0,x_0)} \\
r &\leftarrow r_0(\bar{\theta}, y) \\
\end{align*}
\]

Observation \(y_0\)

\[
\begin{align*}
r &\geq 1 \\
\end{align*}
\]

Stop at time \(n\)
Choose decision \(a = a_n(\bar{\theta}, y)\)

\[
\begin{align*}
n &\leftarrow n + 1 \\
y &\leftarrow y_n \\
\bar{\theta} &\leftarrow \Psi_{n-1}(\bar{\theta}, y) \\
r &\leftarrow r_n(\bar{\theta}, y) \\
\end{align*}
\]

Observation \(y_n\)

\[
\begin{align*}
n &\geq N \\
\end{align*}
\]

Choose decision \(a = 0\)
Path-adapted computable strategy

\[
\begin{align*}
  n & \leftarrow 0 \\
  y & \leftarrow y_0 \\
  \theta & \leftarrow \delta(0, x_0) \\
  r & \leftarrow r_0(\theta, y)
\end{align*}
\]

Observation \( y_0 \)

\[
\begin{align*}
  r & \leftarrow 1? \\
  \text{yes} & \quad \text{Stop at time } n \quad \text{Choose decision } a = a_n(\theta, y) \\
  \text{no} & \\
  n & \leftarrow n + 1 \\
  y & \leftarrow y_n \\
  \theta & \leftarrow \Psi_{n-1}(\theta, y) \\
  r & \leftarrow r_n(\theta, y)
\end{align*}
\]

Observation \( y_n \)

\[
\begin{align*}
  n & \leftarrow n + 1 \\
  y & \leftarrow y_n \\
  \theta & \leftarrow \Psi_{n-1}(\theta, y) \\
  r & \leftarrow r_n(\theta, y)
\end{align*}
\]

Choose decision \( a = 0 \)
Outline

Motivation: Stochastic control

Change-point detection problem

Numerical approximation

Simulation study
- Linear model
- Non linear model

Conclusion and perspectives
Flat/exponential model

- $d = 3$, $p_i = 1/3$, $x_0 = 1$
- $\Phi_0(x, t) = x$, $\Phi_1(x, t) = xe^{0.1t}$, $\Phi_2(x, t) = xe^{0.5t}$, $\Phi_3(x, t) = xe^{1t}$
- $\beta = 1$ (late detection), $\gamma = 1.5$ (wrong mode), $\delta = 1/6$
Flat/exponential model

- $d = 3$, $p_i = 1/3$, $x_0 = 1$
- $\Phi_0(x, t) = x$, $\Phi_1(x, t) = xe^{0.1t}$, $\Phi_2(x, t) = xe^{0.5t}$, $\Phi_3(x, t) = xe^{1t}$
- $\beta = 1$ (late detection), $\gamma = 1.5$ (wrong mode), $\delta = 1/6$
Flat/exponential model

- \( d = 3, \ p_i = 1/3, \ x_0 = 1 \)
- \( \Phi_0(x, t) = x, \ \Phi_1(x, t) = xe^{0.1t}, \ \Phi_2(x, t) = xe^{0.5t}, \ \Phi_3(x, t) = xe^{1t} \)
- \( \beta = 1 \) (late detection), \( \gamma = 1.5 \) (wrong mode), \( \delta = 1/6 \)
Flat/exponential model

- $d = 3$, $p_i = 1/3$, $x_0 = 1$
- $\Phi_0(x, t) = x$, $\Phi_1(x, t) = xe^{0.1t}$, $\Phi_2(x, t) = xe^{0.5t}$, $\Phi_3(x, t) = xe^{1t}$
- $\beta = 1$ (late detection), $\gamma = 1.5$ (wrong mode), $\delta = 1/6$
Flat/exponential model

- $d = 3$, $p_i = 1/3$, $x_0 = 1$
- $\Phi_0(x, t) = x$, $\Phi_1(x, t) = xe^{0.1t}$, $\Phi_2(x, t) = xe^{0.5t}$, $\Phi_3(x, t) = xe^{1t}$
- $\beta = 1$ (late detection), $\gamma = 1.5$ (wrong mode), $\delta = 1/6$
Flat/exponential model

- $d = 3$, $p_i = 1/3$, $x_0 = 1$
- $\Phi_0(x, t) = x$, $\Phi_1(x, t) = xe^{0.1t}$, $\Phi_2(x, t) = xe^{0.5t}$, $\Phi_3(x, t) = xe^{1t}$
- $\beta = 1$ (late detection), $\gamma = 1.5$ (wrong mode), $\delta = 1/6$

<table>
<thead>
<tr>
<th></th>
<th>MA</th>
<th>KF</th>
<th>PDMP</th>
</tr>
</thead>
<tbody>
<tr>
<td>linear link function $F(x) = x$</td>
<td>1.42</td>
<td>1.60</td>
<td>1.00</td>
</tr>
<tr>
<td>inverse link function $F(x) = 1/x$</td>
<td>2.17</td>
<td>1.81</td>
<td>1.17</td>
</tr>
</tbody>
</table>
Non-linear model

- $d = 1$, $x_0 = (0, 0)$
- $\Phi_0((x, u), t) = (\sin(3\pi(u + t)), u + t)$,
  $\Phi_1((x, u), t) = (\sin(5\pi(u + t)), u + t)$
- $\delta = 1/6$, noise variance 1
Non-linear model

- $d = 1$, $x_0 = (0, 0)$
- $\Phi_0((x, u), t) = (\sin(3\pi(u + t)), u + t)$
- $\Phi_1((x, u), t) = (\sin(5\pi(u + t)), u + t)$
- $\delta = 1/6$, noise variance 1

Boxplot of time to jump detection for different values of $\delta$ over 10000 simulation
Outline

- Motivation: Stochastic control
- Change-point detection problem
- Numerical approximation
- Simulation study
- Conclusion and perspectives
Conclusion and perspectives

- Change-point detection method for continuous-time jump dynamics, able to detect a jump and select the post-jump mode
- For general flows but dimension 1
Conclusion and perspectives

- Change-point detection method for continuous-time jump dynamics, able to detect a jump and select the post-jump mode
- For general flows but dimension 1

To be done
- Real data applications
- Theoretical validity of the stopping rule
- Allow to stop between observations
Conclusion and perspectives

- Change-point detection method for continuous-time jump dynamics, able to detect a jump and select the post-jump mode
- For general flows but dimension 1

To be done

- Real data applications
- Theoretical validity of the stopping rule
- Allow to stop between observations
- Several jumps and detections
- Impulse control: select an action that changes the dynamics
- Optimally decide the next observation date
Reference

[BL 17] N. Bäuerle, D. Lange *Optimal control of partially observed PDMPs*

[BdSD 12] A. Brandejsky, B. de Saporta, F. Dufour *Optimal stopping for partially observed PDMPs*

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

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

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

[Geeraert 17] A. Geeraert *Contrôle optimal stochastique des processus de Markov déterministes par morceaux et application à l’optimisation de maintenance* PhD thesis

[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*


[PRS 05] H. Pham, W. Runggaldier, A. Sellami *Approximation by quantization of the filter process and applications to optimal stopping problems under partial observation*