

Un algorithme de cubature pour les EDS et EDSPR de champ moyen

Paul-Eric Chaudru de Raynal

travail en collaboration avec C.A. García Trillos

Université de Nice Sophia-Antipolis, Laboratoire J.A. Dieudonné

Journées MAS,

27 août 2014

EDSPR de champ moyen, ou de McKean-Vlasov (faiblement couplées)

Soit $T > 0$, on considère, sur $[0, T]$, le système

$$\begin{cases} X_t = x + \int_0^t b(s, X_s, \mu_s) ds + \int_0^t \sigma(s, X_s, \mu_s) dB_s, \\ Y_t = \phi(X_T) - \int_t^T H(s, X_s, Y_s, Z_s, \mu_s^{X, Y}) ds + \int_t^T Z_s dB_s \end{cases}$$

EDSPR de champ moyen, ou de McKean-Vlasov (faiblement couplées)

Soit $T > 0$, on considère, sur $[0, T]$, le système

$$\begin{cases} X_t = x + \int_0^t b(s, X_s, \mu_s) ds + \sigma(s, X_s, \mu_s) dB_s, \\ Y_t = \phi(X_T) - \int_t^T H(s, X_s, Y_s, Z_s, \mu_s^{X,Y}) ds + \int_t^T Z_s dB_s \end{cases}$$

► Système de M joueurs

$$\begin{aligned} d\tilde{X}_t^1 &= \tilde{b}(t, \tilde{X}_t^1, \mu_t^M) dt + \tilde{\sigma}(t, \tilde{X}_t^1, \mu_t^M) dB_t^1 \\ &\vdots \\ d\tilde{X}_t^M &= \tilde{b}(t, \tilde{X}_t^M, \mu_t^M) dt + \tilde{\sigma}(t, \tilde{X}_t^M, \mu_t^M) dB_t^M \end{aligned}$$

$$\text{ou } \mu_t^M = \frac{1}{M} \sum_{j=1}^M \delta_{\tilde{X}_t^j}$$

EDSPR de champ moyen, ou de McKean-Vlasov (faiblement couplées)

Soit $T > 0$, on considère, sur $[0, T]$, le système

$$\begin{cases} X_t = x + \int_0^t b(s, X_s, \mu_s) ds + \sigma(s, X_s, \mu_s) dB_s, \\ Y_t = \phi(X_T) - \int_t^T H(s, X_s, Y_s, Z_s, \mu_s^{X, Y}) ds + \int_t^T Z_s dB_s \end{cases}$$

- ▶ Système de M joueurs + un *joueur marqué*

$$d\tilde{X}_t^1 = \tilde{b}(t, \tilde{X}_t^1, \mu_t^M) dt + \tilde{\sigma}(t, \tilde{X}_t^1, \mu_t^M) dB_t^1$$

\vdots

$$d\tilde{X}_t^M = \tilde{b}(t, \tilde{X}_t^M, \mu_t^M) dt + \tilde{\sigma}(t, \tilde{X}_t^M, \mu_t^M) dB_t^M$$

$$d\bar{X}_t^\alpha = b'(t, \bar{X}_t^\alpha, \mu_t^M, \alpha_t) dt + \sigma'(t, \bar{X}_t^\alpha, \mu_t^M) d\bar{B}_t, \quad \bar{X}_0^\alpha = \bar{x}$$

$$\text{ou } \mu_t^M = \frac{1}{M} \sum_{j=1}^M \delta_{\tilde{X}_t^j}, \quad \alpha \text{ a control}$$

- ▶ Fonctionnelle de coût : $\inf_{\alpha \in \mathcal{A}} J(t, \bar{x}, \alpha) = \inf_{\alpha \in \mathcal{A}} \mathbb{E} \left[\phi(\bar{X}_T^{\alpha; t, \bar{x}}, \mu_T^M) + \int_0^T h(s, \bar{X}_s^{\alpha; t, \bar{x}}, \mu_s^M) ds \right]$

EDSPR de champ moyen, ou de McKean-Vlasov (faiblement couplées)

Soit $T > 0$, on considère, sur $[0, T]$, le système

$$\begin{cases} X_t = x + \int_0^t b(s, X_s, \mu_s) ds + \sigma(s, X_s, \mu_s) dB_s, \\ Y_t = \phi(X_T) - \int_t^T H(s, X_s, Y_s, Z_s, \mu_s^{X, Y}) ds + \int_t^T Z_s dB_s \end{cases}$$

- ▶ Système de M joueurs + un *joueur marqué*

$$d\tilde{X}_t^1 = \tilde{b}(t, \tilde{X}_t^1, \mu_t^M) dt + \tilde{\sigma}(t, \tilde{X}_t^1, \mu_t^M) dB_t^1$$

\vdots

$$d\tilde{X}_t^M = \tilde{b}(t, \tilde{X}_t^M, \mu_t^M) dt + \tilde{\sigma}(t, \tilde{X}_t^M, \mu_t^M) dB_t^M$$

$$d\bar{X}_t^\alpha = b'(t, \bar{X}_t^\alpha, \mu_t^M, \alpha_t) dt + \sigma'(t, \bar{X}_t^\alpha, \mu_t^M) d\bar{B}_t, \quad \bar{X}_0^\alpha = \bar{x}$$

$$\text{ou } \mu_t^M = \frac{1}{M} \sum_{j=1}^M \delta_{\tilde{X}_t^j}, \quad \alpha \text{ a control}$$

- ▶ Fonctionnelle de coût : $\inf_{\alpha \in \mathcal{A}} J(t, \bar{x}, \alpha) = \inf_{\alpha \in \mathcal{A}} \mathbb{E} \left[\phi(\bar{X}_T^{\alpha; t, \bar{x}}, \mu_T^M) + \int_0^T h(s, \bar{X}_s^{\alpha; t, \bar{x}}, \mu_s^M) ds \right]$
- ▶ **Problème possiblement difficile pour M grand**
↳ Approximation asymptotique? (Lasry, Lions)

EDSPR de champ moyen, ou de McKean-Vlasov (faiblement couplées)

Soit $T > 0$, on considère, sur $[0, T]$, le système

$$\begin{cases} X_t = x + \int_0^t b(s, X_s, \mu_s) ds + \sigma(s, X_s, \mu_s) dB_s, \\ Y_t = \phi(X_T) - \int_t^T H(s, X_s, Y_s, Z_s, \mu_s^{X, Y}) ds + \int_t^T Z_s dB_s \end{cases}$$

- ▶ Système de M joueurs + un *joueur marqué*

$$d\tilde{X}_t^1 = \tilde{b}(t, \tilde{X}_t^1, \mu_t) dt + \tilde{\sigma}(t, \tilde{X}_t^1, \mu_t) dB_t^1$$

⋮

$$d\tilde{X}_t^m = \tilde{b}(t, \tilde{X}_t^m, \mu_t) dt + \tilde{\sigma}(t, \tilde{X}_t^m, \mu_t) dB_t^m$$

⋮

$$d\bar{X}_t^\alpha = b'(t, \bar{X}_t^\alpha, \mu_t, \alpha_t) dt + \sigma'(t, \bar{X}_t^\alpha, \mu_t) d\bar{B}_t, \quad \bar{X}_0^\alpha = \bar{x}$$

$$\text{ou } \mu_t^M = \frac{1}{M} \sum_{j=1}^M \delta_{\tilde{X}_t^j}, \quad \alpha \text{ a control}$$

- ▶ $M \rightarrow +\infty$ $\xrightarrow{\text{law of large number}} \mu^M \rightarrow \mu, \quad X = (\tilde{X}^1, \bar{X})^*$

- ▶ Fonctionnelle de coût : $\inf_{\alpha \in \mathcal{A}} J(t, \bar{x}, \alpha) = \inf_{\alpha \in \mathcal{A}} \mathbb{E} \left[\phi(\bar{X}_T^{\alpha; t, \bar{x}}, \mu_T) + \int_0^T h(s, \bar{X}_s^{\alpha; t, \bar{x}}, \mu_t) ds \right]$

- ▶ Solution d'une HJB :

$$(\partial_t + \mathcal{L})u(t, x) = H(t, x, u(t, x), \sigma(x, \mu_t) D_x u(t, x), \mu_t), \quad H = \inf_{\alpha} \{b' + h\}$$

EDSPR de champ moyen, ou de McKean-Vlasov (faiblement couplées)

Soit $T > 0$, on considère, sur $[0, T]$, le système

$$\begin{cases} X_t = x + \int_0^t b(s, X_s, \mu_s) ds + \sigma(s, X_s, \mu_s) dB_s, \\ Y_t = \phi(X_T) - \int_t^T H(s, X_s, Y_s, Z_s, \mu_s^{X, Y}) ds + \int_t^T Z_s dB_s \end{cases}$$

- ▶ Problème de contrôle d'un joueur dans un environnement "champ moyen"
- ▶ On a $Y_t = u(t, X_t)$ ou u est la solution de l'EDP :

$$(\partial_t + \mathcal{L})u(t, x) = H(t, x, u(t, x), \sigma(x, \mu_t) D_x u(t, x), \mu_t^{X, Y}), \quad (t, x) \in [0, T] \times \mathbb{R}^d$$

- ▶ **Objectif : approcher μ_T et u**

EDSPR de champ moyen, ou de McKean-Vlasov (faiblement couplées)

Soit $T > 0$, on considère, sur $[0, T]$, le système

$$\begin{cases} X_t = x + \int_0^t b(s, X_s, \mu_s) ds + \sigma(s, X_s, \mu_s) dB_s, \\ Y_t = \phi(X_T) - \int_t^T H(s, X_s, Y_s, Z_s, \mu_s^{X, Y}) ds + \int_t^T Z_s dB_s \end{cases}$$

- ▶ Problème de contrôle d'un joueur dans un environnement "champ moyen"
- ▶ On a $Y_t = u(t, X_t)$ ou u est la solution de l'EDP :

$$(\partial_t + \mathcal{L})u(t, x) = H(t, x, u(t, x), \sigma(x, \mu_t) D_x u(t, x), \mu_t^{X, Y}), \quad (t, x) \in [0, T] \times \mathbb{R}^d$$

- ▶ **Objectif : approcher μ_T et u**

↳ Approche basée sur **les méthodes de cubature**

Cubature sur l'espace de Wiener

↳ Lyons, Victoir (2002)

- ▶ $T > 0$, $(\mathcal{C}([0, T], \mathbb{R}), \mathbb{P})$, espace de Wiener.
- ▶ Approcher la mesure de Wiener \mathbb{P} par une mesure discrète \mathbb{Q} à support fini dans l'ensemble des trajectoires à variations bornées telle que :

$$(\mathbb{E}_{\mathbb{P}} - \mathbb{E}_{\mathbb{Q}}) \left[\int_{0=t_0 < t_1 < \dots < t_l = T} \circ dB_{t_1} \cdots \circ dB_{t_l} \right] = 0,$$

pour tout $l \leq m$, $m \in \mathbb{N}$ donné.

- ▶ Existence : “Théorème de Tchakaloff”.
- ▶ Pour toute fonctionnelle régulière F :

$$(\mathbb{E}_{\mathbb{P}} - \mathbb{E}_{\mathbb{Q}})[F(B_T)] \leq CT^{(m+1)/2} \sup_{j \leq m+2} \|\nabla_x^j F\|_{\infty}.$$

- ▶ L'EDS est remplacée par un système d'EDO pondérées.

Cadre de travail

Soit $T > 0$, on considère sur $[0, T]$ le système uni dimensionnel suivant :

$$\begin{cases} dX_t^x = V(X_t^x, \mu_t) \circ dB_t \\ dY_t^x = -f(t, X_t^x, Y_t^x, Z_t^x, \mu_t^{X,Y})dt + Z_t^x dB_t \\ X_0^x = x, \quad Y_T^x = \phi(X_T^x), \end{cases}$$

► Approche basée sur les méthodes de cubature :

Les coefficients V et f sont réguliers en espace, et au moins Lipschitz par rapport à la mesure

► Deux étapes :

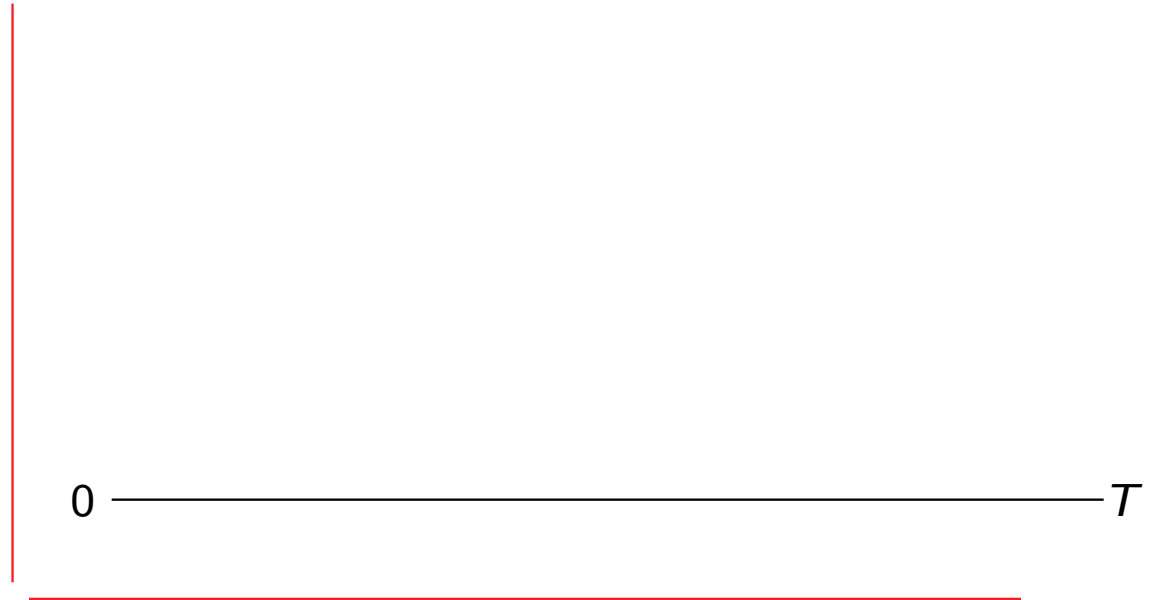
1. construire un arbre de cubature $\mathcal{T}(m)$ (m est l'ordre de la cubature) qui approche la loi de la composante progressive
2. remonter l'arbre à rebours pour calculer les valeurs de la composante rétrograde (Y, Z)

Algorithme, première étape : construction de l'arbre de cubature $\mathcal{T}(m)$

Initialisation :

- ▶ Intervalle $[0, T]$,

exemple : cubature d'ordre 3, $N = 3$.

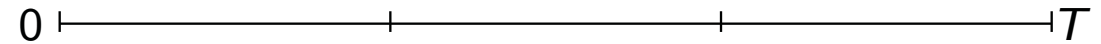


Algorithme, première étape : construction de l'arbre de cubature $\mathcal{T}(m)$

Initialisation :

- ▶ Intervalle $[0, T]$, $N > 0$
- ▶ $T_0 = 0 < \dots < T_N = T$
- ▶ Cubature d'ordre m :
 $\{w_t^1, \dots, w_t^n\}, \{\lambda_1, \dots, \lambda_n\}$

exemple : cubature d'ordre 3, $N = 3$.



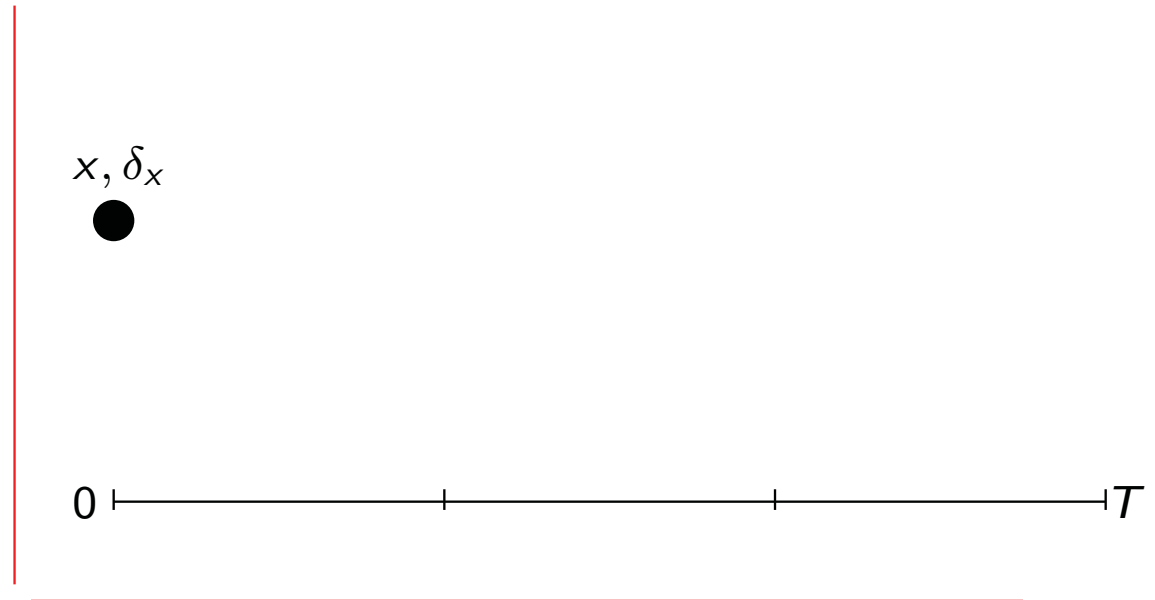
Algorithme, première étape : construction de l'arbre de cubature $\mathcal{T}(m)$

Initialisation :

- ▶ Intervalle $[0, T]$, $N > 0$
- ▶ $T_0 = 0 < \dots < T_N = T$
- ▶ Cubature d'ordre m :
 $\{w_t^1, \dots, w_t^n\}, \{\lambda_1, \dots, \lambda_n\}$

- ▶ Vraie dynamique : (x, δ_x)
- ▶ Approximation : (x, δ_x)

exemple : cubature d'ordre 3, $N = 3$.



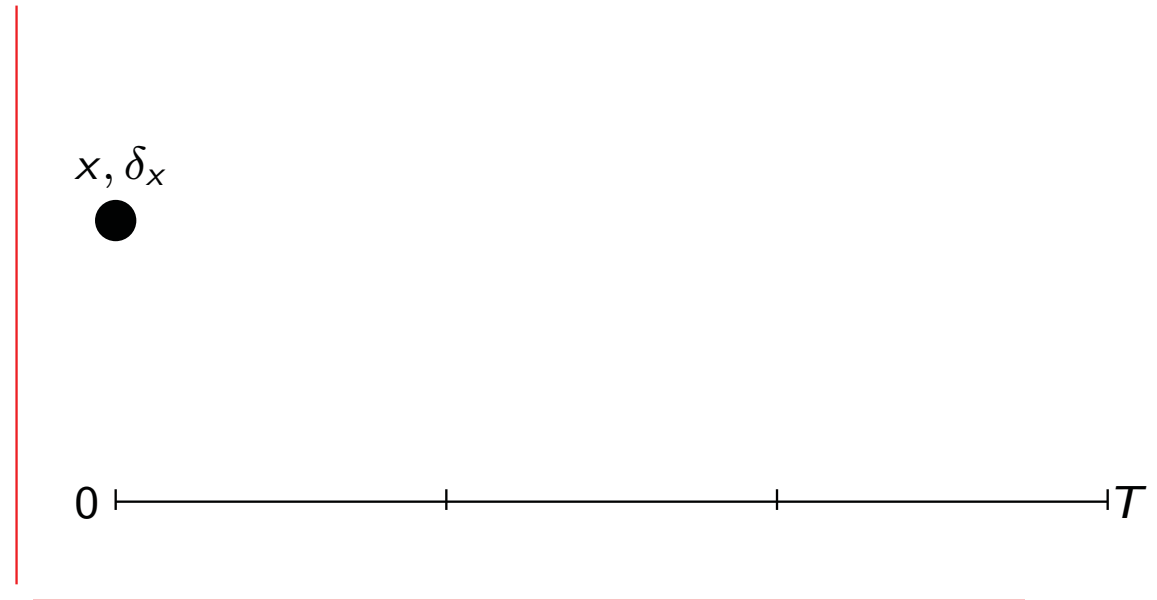
Algorithme, première étape : construction de l'arbre de cubature $\mathcal{T}(m)$

étape $[0, T_1[$

- ▶ Intervalle $[0, T]$, $N > 0$
- ▶ $T_0 = 0 < \dots < T_N = T$
- ▶ Cubature d'ordre m :

$$\{w_t^1, \dots, w_t^n\}, \{\lambda_1, \dots, \lambda_n\}$$

exemple : cubature d'ordre 3, $N = 3$.



- ▶ Vraie dynamique : $dX_t = V(X_t, \mu_t) \circ dB_t$
- ▶ Approximation : $d\hat{X}_t^{(j)} = V(\hat{X}_t^{(j)}, ?_{T_0})dw_t^j, \quad \hat{X}_{T_1}^{(j)} = \hat{X}_{T_1}^{\delta_x} \quad j = 1, \dots, n$

Algorithme, première étape : construction de l'arbre de cubature $\mathcal{T}(m)$

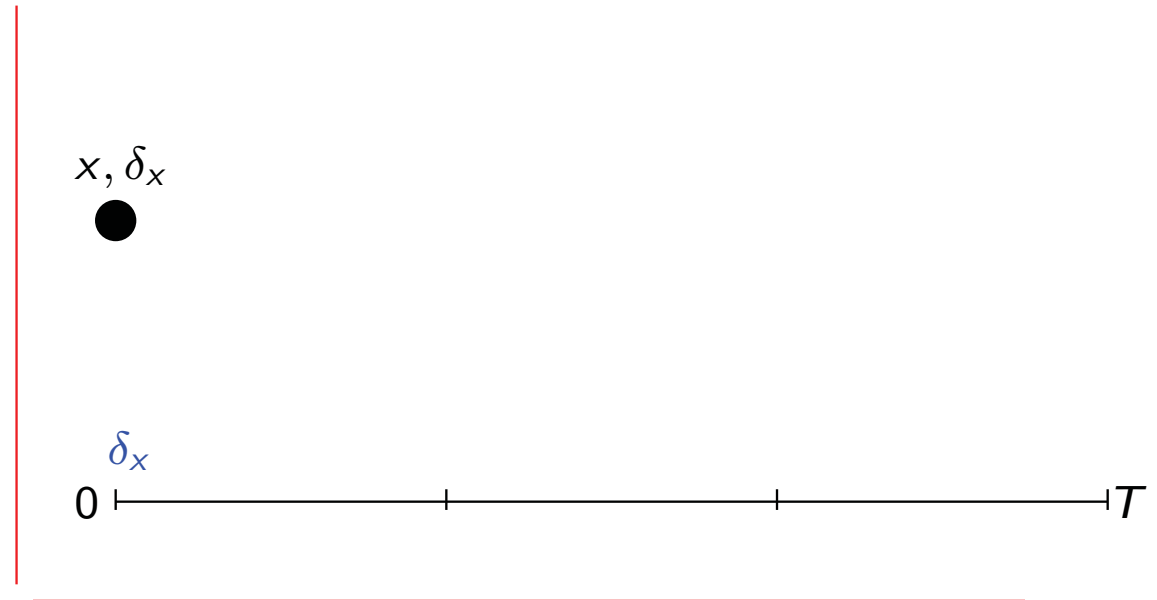
étape $[0, T_1[$

- ▶ Intervalle $[0, T]$, $N > 0$
- ▶ $T_0 = 0 < \dots < T_N = T$
- ▶ Cubature d'ordre m :

$$\{w_t^1, \dots, w_t^n\}, \{\lambda_1, \dots, \lambda_n\}$$

- ▶ Vraie dynamique : $dX_t = V(X_t, \mu_t) \circ dB_t$
- ▶ Approximation : $d\hat{X}_t^{(j)} = V(\hat{X}_t^{(j)}, \delta_x)dw_t^j, \quad \hat{X}_{T_1}^{(j)} = \hat{X}_{T_1}^{\delta_x} \quad j = 1, \dots, n$

exemple : cubature d'ordre 3, $N = 3$.

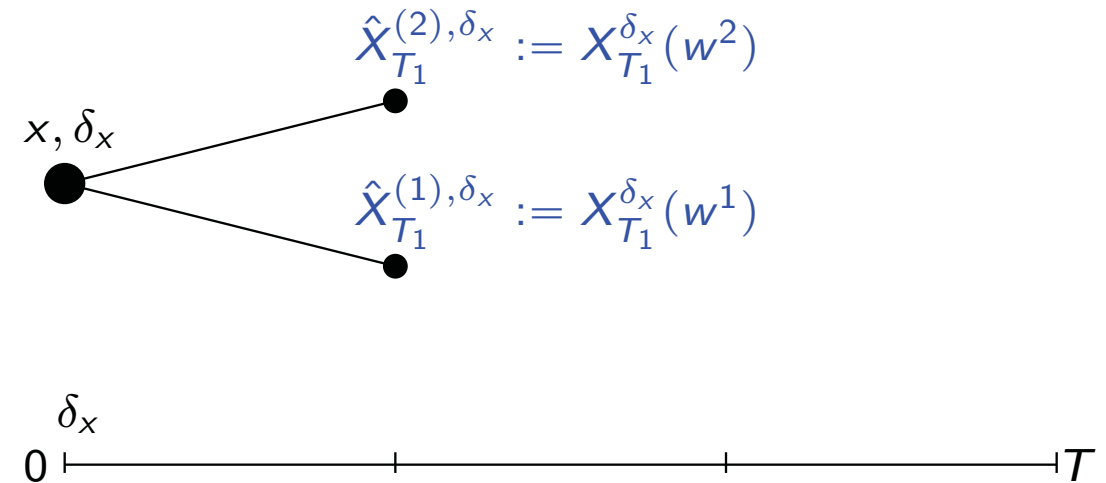


Algorithme, première étape : construction de l'arbre de cubature $\mathcal{T}(m)$

étape $[0, T_1[$

- ▶ Intervalle $[0, T]$, $N > 0$
- ▶ $T_0 = 0 < \dots < T_N = T$
- ▶ Cubature d'ordre m :
 $\{w_t^1, \dots, w_t^n\}, \{\lambda_1, \dots, \lambda_n\}$

exemple : cubature d'ordre 3, $N = 3$.



- ▶ Vraie dynamique : $dX_t = V(X_t, \mu_t) \circ dB_t$
- ▶ Approximation : $d\hat{X}_t^{(j)} = V(\hat{X}_t^{(j)}, \delta_x)dw_t^j, \quad \hat{X}_{T_1}^{(j)} = \hat{X}_{T_1}^{\delta_x} \quad j = 1, \dots, n$

Algorithme, première étape : construction de l'arbre de cubature $\mathcal{T}(m)$

étape $[T_1, T_2[$

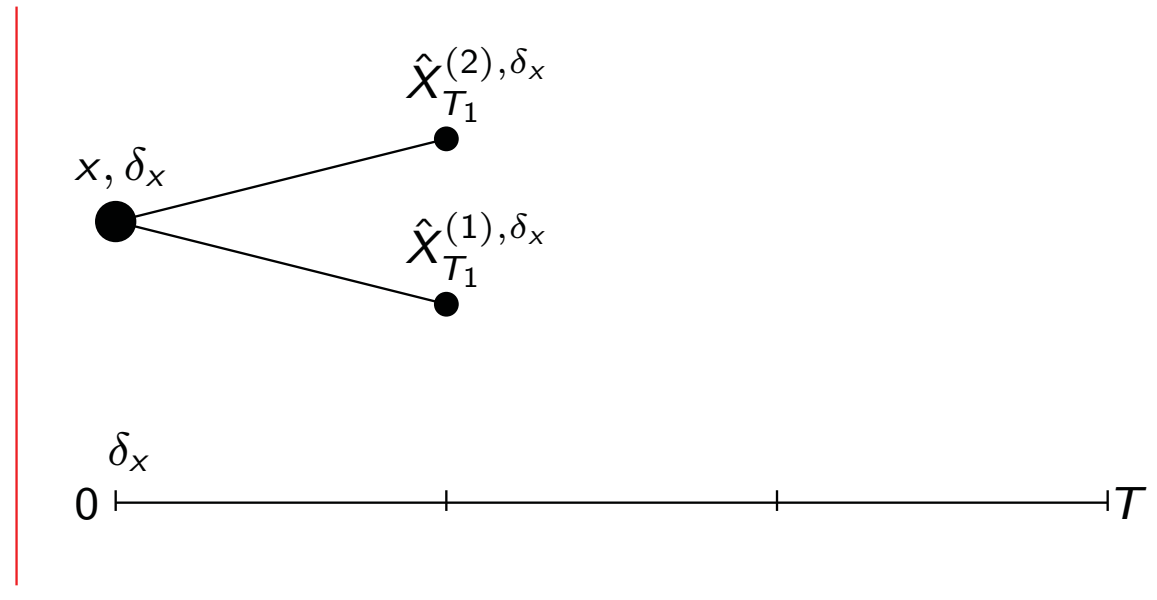
- ▶ Intervalle $[0, T]$, $N > 0$
- ▶ $T_0 = 0 < \dots < T_N = T$
- ▶ Cubature d'ordre m :

$$\{w_t^1, \dots, w_t^n\}, \{\lambda_1, \dots, \lambda_n\}$$

▶ Vraie dynamique : $dX_t = V(X_t, \mu_t) \circ dB_t$

▶ Approximation : $d\hat{X}_t^{(k,j)} = V(\hat{X}_t^{(k,j)}, \lambda_t) dw_t^j, \quad \hat{X}_{T_1}^{(k,j)} = \hat{X}_{T_1}^{k, \delta_x} \quad k, j = 1, \dots, n$

exemple : cubature d'ordre 3, $N = 3$.



Algorithme, première étape : construction de l'arbre de cubature $\mathcal{T}(m)$

étape $[T_1, T_2[$

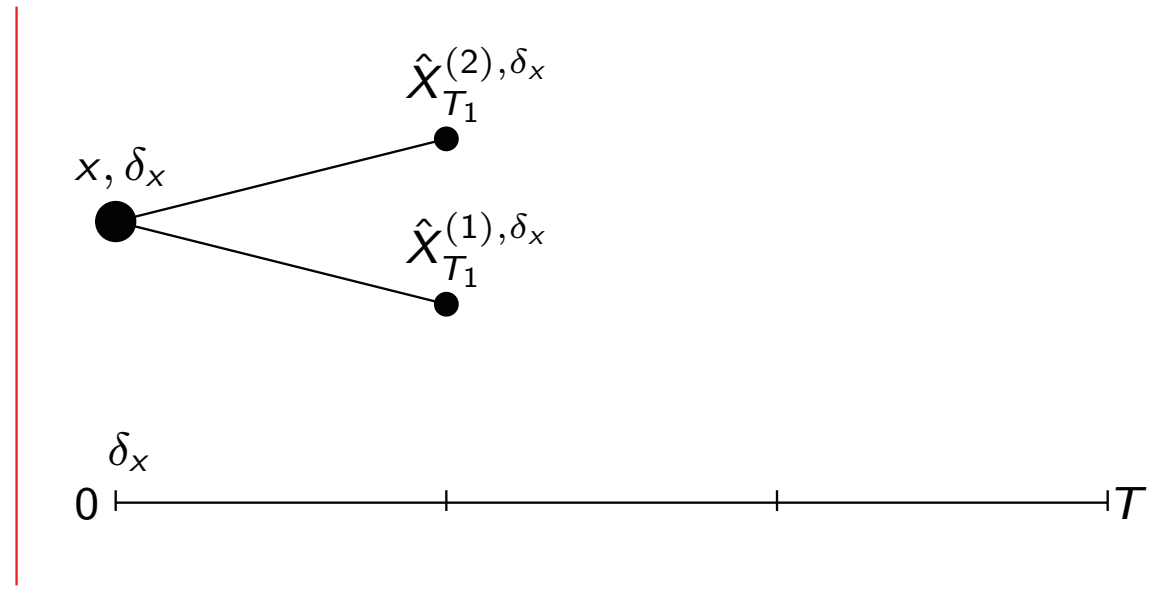
- ▶ Intervalle $[0, T]$, $N > 0$
- ▶ $T_0 = 0 < \dots < T_N = T$
- ▶ Cubature d'ordre m :

$$\{w_t^1, \dots, w_t^n\}, \{\lambda_1, \dots, \lambda_n\}$$

- ▶ Vraie dynamique : $dX_t = V(X_t, \mu_t) \circ dB_t$

- ▶ Approximation : $d\hat{X}_t^{(k,j)} = V(\hat{X}_t^{(k,j)}, ??_{T_1})dw_t^j, \quad \hat{X}_{T_1}^{(k,j)} = \hat{X}_{T_1}^{k, \delta_x} \quad k, j = 1, \dots, n$

exemple : cubature d'ordre 3, $N = 3$.



Algorithme, première étape : construction de l'arbre de cubature $\mathcal{T}(m)$

étape $[T_1, T_2[$

- ▶ Intervalle $[0, T]$, $N > 0$
- ▶ $T_0 = 0 < \dots < T_N = T$
- ▶ Cubature d'ordre m :

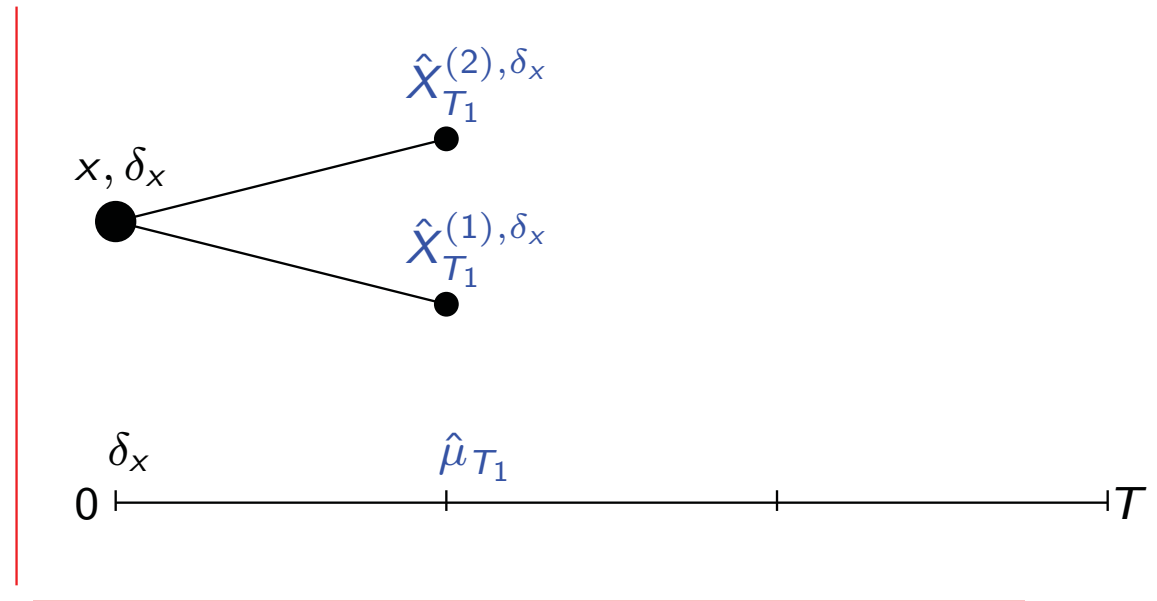
$$\{w_t^1, \dots, w_t^n\}, \{\lambda_1, \dots, \lambda_n\}$$

- ▶ Vraie dynamique : $dX_t = V(X_t, \mu_t) \circ dB_t$

- ▶ Approximation : $d\hat{X}_t^{(k,j)} = V(\hat{X}_t^{(k,j)}, \hat{\mu}_{T_1}) dw_t^j, \quad \hat{X}_{T_1}^{(k,j)} = \hat{X}_{T_1}^{k, \delta_x} \quad k, j = 1, \dots, n$

- ▶ $\hat{\mu}_{T_1} = \sum_{j=1}^n \lambda_j \delta_{\hat{X}_{T_1}^{(j), \delta_x}}$

exemple : cubature d'ordre 3, $N = 3$.



Algorithme, première étape : construction de l'arbre de cubature $\mathcal{T}(m)$

étape $[T_1, T_2[$

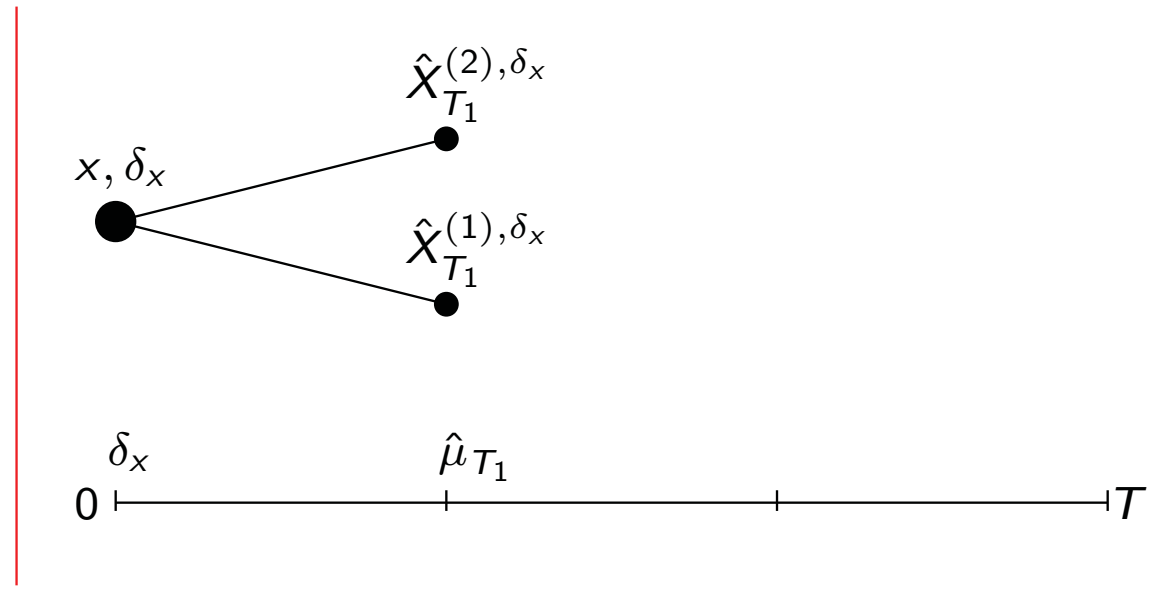
- ▶ Intervalle $[0, T]$, $N > 0$
- ▶ $T_0 = 0 < \dots < T_N = T$
- ▶ Cubature d'ordre m :

$$\{w_t^1, \dots, w_t^n\}, \{\lambda_1, \dots, \lambda_n\}$$

- ▶ Vraie dynamique : $dX_t = V(X_t, \mu_t) \circ dB_t$
- ▶ Approximation : $d\hat{X}_t^{(k,j)} = V(\hat{X}_t^{(k,j)}, \hat{\mu}_{T_1}) dw_t^j, \quad \hat{X}_{T_1}^{(k,j)} = \hat{X}_{T_1}^{k, \delta_x} \quad k, j = 1, \dots, n$

- ▶
$$\hat{\mu}_{T_1} = \sum_{j=1}^n \lambda_j \delta_{\hat{X}_{T_1}^{(j), \delta_x}}$$

exemple : cubature d'ordre 3, $N = 3$.



Algorithme, première étape : construction de l'arbre de cubature $\mathcal{T}(m)$

étape $[T_1, T_2[$

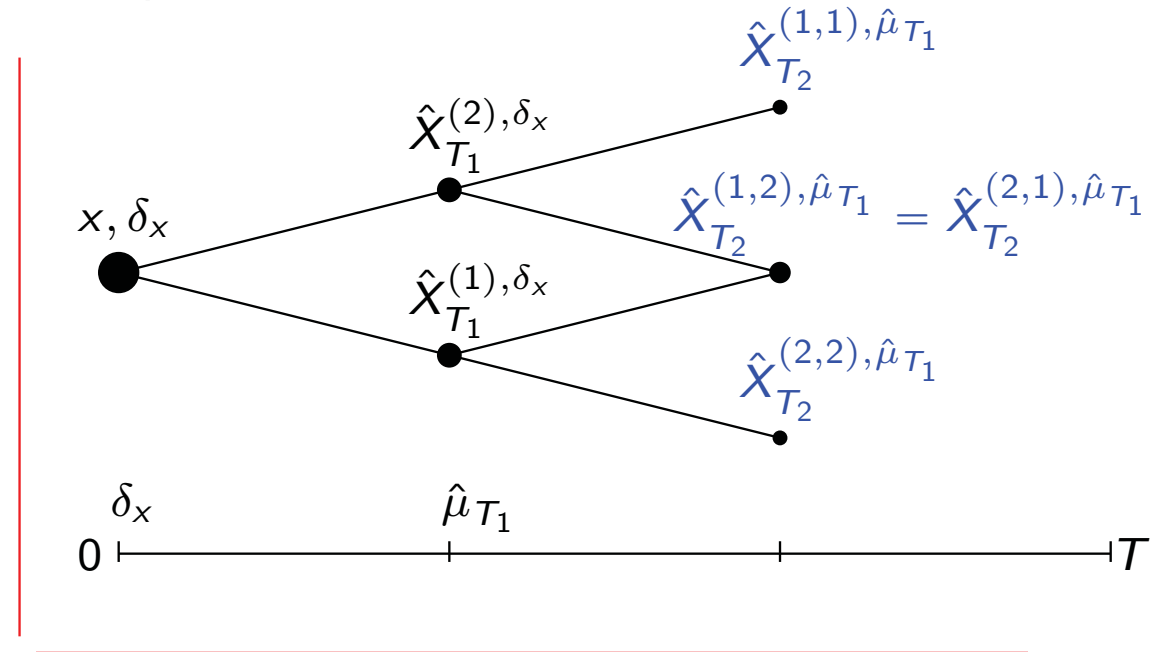
- ▶ Intervalle $[0, T]$, $N > 0$
- ▶ $T_0 = 0 < \dots < T_N = T$
- ▶ Cubature d'ordre m :

$$\{w_t^1, \dots, w_t^n\}, \{\lambda_1, \dots, \lambda_n\}$$

- ▶ Vraie dynamique : $dX_t = V(X_t, \mu_t) \circ dB_t$
- ▶ Approximation : $d\hat{X}_t^{(k,j)} = V(\hat{X}_t^{(k,j)}, \hat{\mu}_{T_1}) dw_t^j, \quad \hat{X}_{T_1}^{(k,j)} = \hat{X}_{T_1}^{k, \delta_x} \quad k, j = 1, \dots, n$

$$\hat{\mu}_{T_1} = \sum_{j=1}^n \lambda_j \delta_{\hat{X}_{T_1}^{(j), \delta_x}}$$

exemple : cubature d'ordre 3, $N = 3$.



Algorithme, première étape : construction de l'arbre de cubature $\mathcal{T}(m)$

étape $[T_1, T_2[$

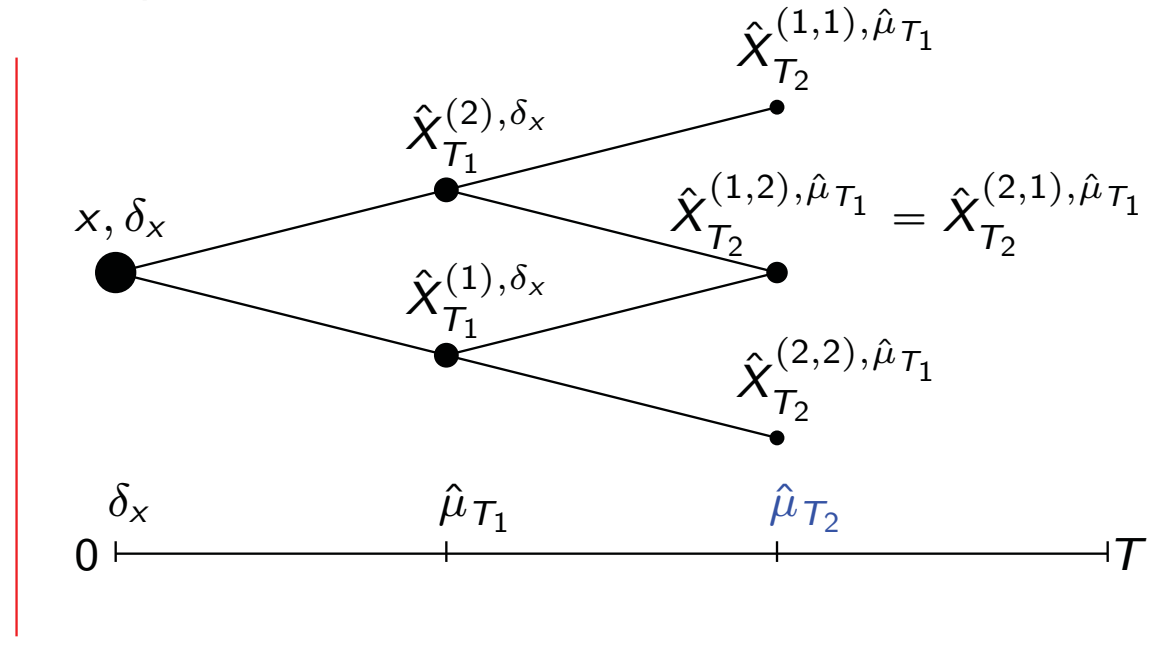
- ▶ Intervalle $[0, T]$, $N > 0$
- ▶ $T_0 = 0 < \dots < T_N = T$
- ▶ Cubature d'ordre m :

$$\{w_t^1, \dots, w_t^n\}, \{\lambda_1, \dots, \lambda_n\}$$

- ▶ Vraie dynamique : $dX_t = V(X_t, \mu_t) \circ dB_t$
- ▶ Approximation : $d\hat{X}_t^{(k,j)} = V(\hat{X}_t^{(k,j)}, \hat{\mu}_{T_1}) dw_t^j, \quad \hat{X}_{T_1}^{(k,j)} = \hat{X}_{T_1}^{k, \delta_x} \quad k, j = 1, \dots, n$

$$\hat{\mu}_{T_2} = \sum_{i,j=1}^n \lambda_i \lambda_j \delta_{\hat{X}_{T_2}^{(i,j), \hat{\mu}}}$$

exemple : cubature d'ordre 3, $N = 3$.



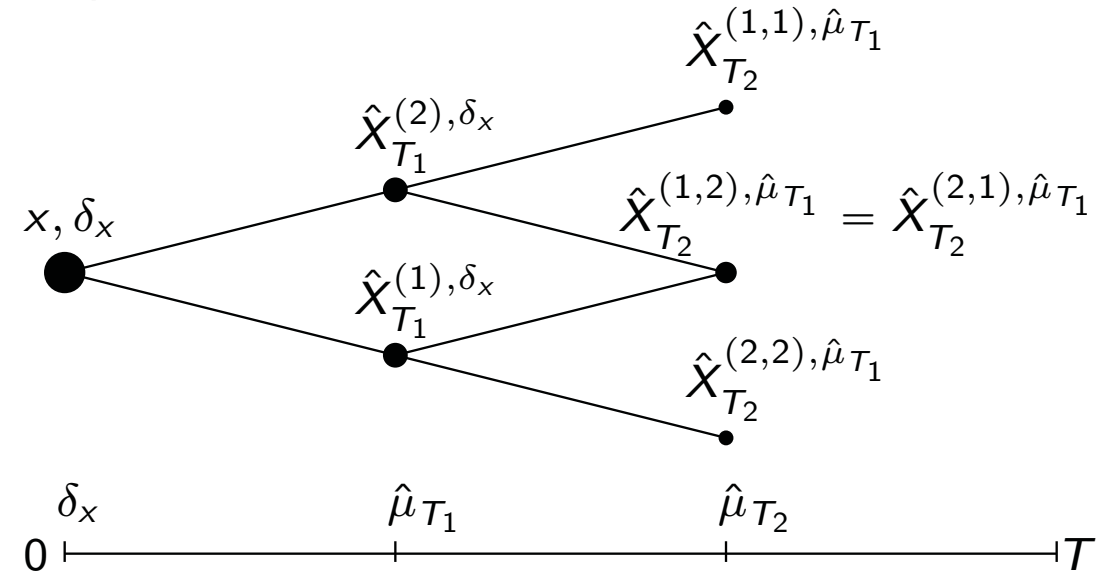
Algorithme, première étape : construction de l'arbre de cubature $\mathcal{T}(m)$

étape $[T_k, T_{k+1}[$

- ▶ Intervalle $[0, T]$, $N > 0$
- ▶ $T_0 = 0 < \dots < T_N = T$
- ▶ Cubature d'ordre m :

$$\{w_t^1, \dots, w_t^n\}, \{\lambda_1, \dots, \lambda_n\}$$

exemple : cubature d'ordre 3, $N = 3$.



▶ Vraie dynamique : $dX_t = V(X_t, \mu_t) \circ dB_t$

▶ Approximation : $d\hat{X}_t^{(\pi,j)} = V(\hat{X}_t^{(\pi,j)}, \hat{\mu}_{T_k})dw_t^j$, $\hat{X}_{T_1}^{(\pi,j)} = \hat{X}_{T_1}^{\pi, \delta_x}$ $j = 1, \dots, n$
 $\pi \in \{1, \dots, n\}^k$

$$\hat{\mu}_{T_k} = \sum_{\pi \in \{1, \dots, n\}^k} \left(\prod_{i_j \in \pi} \lambda_{i_j} \right) \delta_{\hat{X}_{T_k}^{\pi, \hat{\mu}}}$$

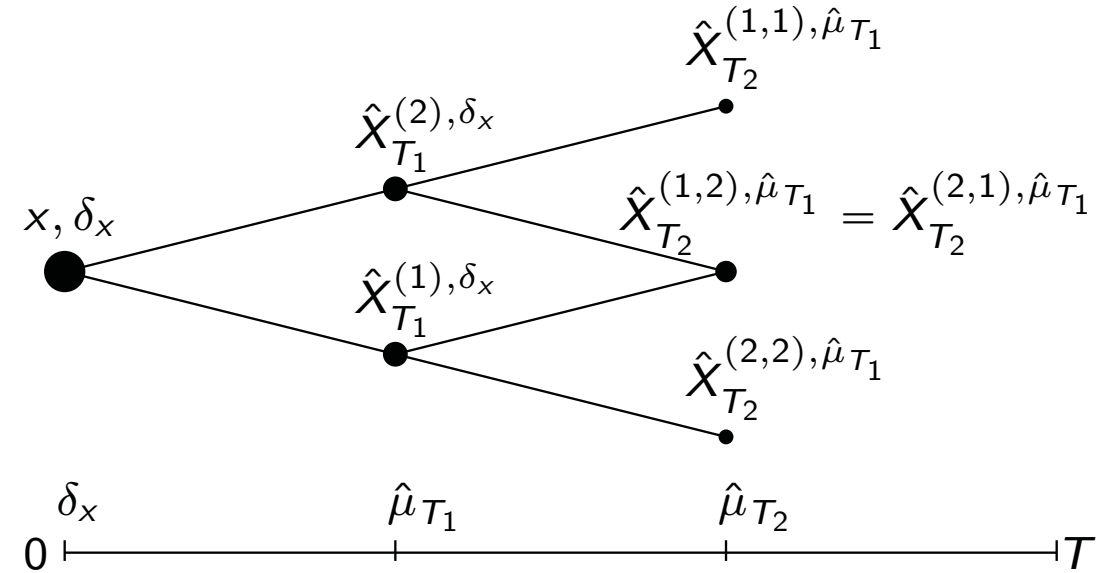
Algorithme, première étape : construction de l'arbre de cubature $\mathcal{T}(m)$

étape $[T_k, T_{k+1}[$

- ▶ Intervalle $[0, T]$, $N > 0$
- ▶ $T_0 = 0 < \dots < T_N = T$
- ▶ Cubature d'ordre m :

$$\{w_t^1, \dots, w_t^n\}, \{\lambda_1, \dots, \lambda_n\}$$

exemple : cubature d'ordre 3, $N = 3$.



▶ Vraie dynamique : $dX_t = V(X_t, \mu_t) \circ dB_t$

▶ Approximation : $d\hat{X}_t^{(\pi,j)} = V(\hat{X}_t^{(\pi,j)}, \hat{\mu}_{T_k})dw_t^j, \quad \hat{X}_{T_1}^{(\pi,j)} = \hat{X}_{T_1}^{\pi, \delta_x} \quad j = 1, \dots, n$
 $\pi \in \{1, \dots, n\}^k$

▶ $\hat{\mu}_{T_k} = \sum_{\pi \in \{1, \dots, n\}^k} \left(\prod_{i_j \in \pi} \lambda_{i_j} \right) \delta_{\hat{X}_{T_k}^{\pi, \hat{\mu}}}$

Erreur (faible) \mathcal{T}

- ▶ Objectif : contrôler $|\langle \mu_{\mathcal{T}} - \hat{\mu}_{\mathcal{T}}, \phi \rangle|$
 - ▶ Propriété de Markov : **erreur globale = somme des erreurs locales**
- ▶ erreur locale $[T_{k-1}, T_k]$: $E_{T_k} = |\langle \mu_{T_k} - \hat{\mu}_{T_k}, \psi \rangle|$, ψ fonction régulière .

$$\begin{array}{ll} \text{Vraie dynamique :} & dX_t = V(X_t, \mu_t) \circ dB_t \\ \text{Approximation :} & d\hat{X}_t^{(j), \hat{\mu}} = V(\hat{X}_t^{(j), \hat{\mu}}, \hat{\mu}_{T_{k-1}}) dw_t^j, \quad j = 1, \dots, n, \end{array} \quad \begin{array}{l} X_{T_{k-1}} = y \\ \hat{X}_{T_{k-1}}^{(j), \hat{\mu}} = y \end{array}$$

$$E_{T_k} \leq ?$$

Erreur (faible) \mathcal{T}

- ▶ **Objectif : contrôler** $|\langle \mu_{\mathcal{T}} - \hat{\mu}_{\mathcal{T}}, \phi \rangle|$
 - ▶ Propriété de Markov : **erreur globale = somme des erreurs locales**
- ▶ **erreur locale** $[T_{k-1}, T_k]$: $E_{T_k} = |\langle \mu_{T_k} - \hat{\mu}_{T_k}, \psi \rangle|$, ψ fonction régulière .

Vraie dynamique : $dX_t = V(X_t, \mu_t) \circ dB_t$ $X_{T_{k-1}} = y$

Approximation : $d\hat{X}_t^{(j), \hat{\mu}} = V(\hat{X}_t^{(j), \hat{\mu}}, \hat{\mu}_{T_{k-1}})dw_t^j$, $j = 1, \dots, n$, $\hat{X}_{T_{k-1}}^{(j), \hat{\mu}} = y$

$$E_{T_k} = \text{erreur Euler} \approx \Delta_{T_k}^{3/2} + \text{propagation d'erreur} \approx \Delta_{T_k} E_{T_{k-1}} + \text{erreur de cubature} \approx \Delta_{T_k}^{(m+1)/2}$$

Erreur (faible) \mathcal{T}

- ▶ **Objectif : contrôler** $|\langle \mu_{\mathcal{T}} - \hat{\mu}_{\mathcal{T}}, \phi \rangle|$
 - ▶ Propriété de Markov : **erreur globale = somme des erreurs locales**
- ▶ **erreur locale** $[T_{k-1}, T_k]$: $E_{T_k} = |\langle \mu_{T_k} - \hat{\mu}_{T_k}, \psi \rangle|$, ψ fonction régulière .

Vraie dynamique : $dX_t = V(X_t, \mu_t) \circ dB_t$ $X_{T_{k-1}} = y$

Approximation : $d\hat{X}_t^{(j), \hat{\mu}} = V(\hat{X}_t^{(j), \hat{\mu}}, \hat{\mu}_{T_{k-1}})dw_t^j$, $j = 1, \dots, n$, $\hat{X}_{T_{k-1}}^{(j), \hat{\mu}} = y$

$E_{T_k} =$ erreur Euler $\approx \Delta_{T_k}^{3/2}$ + propagation d'erreur $\approx \Delta_{T_k} E_{T_{k-1}}$ + erreur de cubature $\approx \Delta_{T_k}^{(m+1)/2}$

Vitesse d'approximation : soit $\mathcal{T}(m)$ un arbre et $T_k = T(1 - (1 - \frac{k}{N}))$, $k = 1, \dots, N$, une subdivision :

- ▶ si ϕ régulière alors

$$|\langle \mu_{\mathcal{T}} - \hat{\mu}_{\mathcal{T}}, \phi \rangle| \leq CN^{-([\frac{m-1}{2}] \wedge 1/2)} \sup\{\|\nabla_x^j \phi\|_{\infty}; j \leq m+2\},$$

Erreur (faible) \mathcal{T}

- ▶ **Objectif : contrôler** $|\langle \mu_T - \hat{\mu}_T, \phi \rangle|$
 - ▶ Propriété de Markov : **erreur globale = somme des erreurs locales**
- ▶ **erreur locale** $[T_{k-1}, T_k]$: $E_{T_k} = |\langle \mu_{T_k} - \hat{\mu}_{T_k}, \psi \rangle|$, ψ fonction régulière .

Vraie dynamique : $dX_t = V(X_t, \mu_t) \circ dB_t$ $X_{T_{k-1}} = y$

Approximation : $d\hat{X}_t^{(j), \hat{\mu}} = V(\hat{X}_t^{(j), \hat{\mu}}, \hat{\mu}_{T_{k-1}})dw_t^j$, $j = 1, \dots, n$, $\hat{X}_{T_{k-1}}^{(j), \hat{\mu}} = y$

$E_{T_k} =$ **erreur Euler** $\approx \Delta_{T_k}^{3/2}$ + propagation d'erreur $\approx \Delta_{T_k} E_{T_{k-1}}$ + **erreur de cubature** $\approx \Delta_{T_k}^{(m+1)/2}$

Vitesse d'approximation : soit $\mathcal{T}(m)$ un arbre et $T_k = T(1 - (1 - \frac{k}{N}))$, $k = 1, \dots, N$, une subdivision :

- ▶ si ϕ régulière alors

$$|\langle \mu_T - \hat{\mu}_T, \phi \rangle| \leq CN^{-([\frac{m-1}{2}] \wedge 1/2)} \sup\{\|\nabla_x^j \phi\|_\infty; j \leq m+2\},$$

Erreur (faible) \mathcal{T}

- ▶ **Objectif : contrôler** $|\langle \mu_{\mathcal{T}} - \hat{\mu}_{\mathcal{T}}, \phi \rangle|$
 - ▶ Propriété de Markov : **erreur globale = somme des erreurs locales**
- ▶ **erreur locale** $[T_{k-1}, T_k]$: $E_{T_k} = |\langle \mu_{T_k} - \hat{\mu}_{T_k}, \psi \rangle|$, ψ fonction régulière .

Vraie dynamique : $dX_t = V(X_t, \mu_t) \circ dB_t$ $X_{T_{k-1}} = y$

Approximation : $d\hat{X}_t^{(j), \hat{\mu}} = V(\hat{X}_t^{(j), \hat{\mu}}, \hat{\mu}_{T_{k-1}}) dw_t^j$, $j = 1, \dots, n$, $\hat{X}_{T_{k-1}}^{(j), \hat{\mu}} = y$

$E_{T_k} =$ **erreur Euler** $\approx \Delta_{T_k}^{3/2}$ + propagation d'erreur $\approx \Delta_{T_k} E_{T_{k-1}}$ + erreur de cubature $\approx \Delta_{T_k}^{(m+1)/2}$

Vitesse d'approximation : soit $\mathcal{T}(m)$ un arbre et $T_k = T(1 - (1 - \frac{k}{N}))$, $k = 1, \dots, N$, une subdivision :

- ▶ si ϕ régulière alors

$$|\langle \mu_{\mathcal{T}} - \hat{\mu}_{\mathcal{T}}, \phi \rangle| \leq CN^{-([\frac{m-1}{2}] \wedge 1/2)} \sup\{\|\nabla_x^j \phi\|_{\infty}; j \leq m+2\},$$

Erreur (faible) \mathcal{T}

- ▶ **Objectif : contrôler** $|\langle \mu_T - \hat{\mu}_T, \phi \rangle|$
 - ▶ Propriété de Markov : **erreur globale = somme des erreurs locales**
- ▶ **erreur locale** $[T_{k-1}, T_k]$: $E_{T_k} = |\langle \mu_{T_k} - \hat{\mu}_{T_k}, \psi \rangle|$, ψ fonction régulière φ régulière.

Vraie dynamique : $dX_t = V(X_t, \langle \mu_t, \varphi \rangle) \circ dB_t$ $X_{T_{k-1}} = y$
Approximation : $d\hat{X}_t^{(j), \hat{\mu}} = V(\hat{X}_t^{(j), \hat{\mu}}, \langle \hat{\mu}_{T_{k-1}}, \varphi \rangle) dw_t^j$, $j = 1, \dots, n$, $\hat{X}_{T_{k-1}}^{(j), \hat{\mu}} = y$

$$E_{T_k} = \text{erreur Euler} \approx \Delta_{T_k}^{3/2} + \text{propagation d'erreur} \approx \Delta_{T_k} E_{T_{k-1}} + \text{erreur de cubature} \approx \Delta_{T_k}^{(m+1)/2}$$

Vitesse d'approximation : soit $\mathcal{T}(m)$ un arbre et $T_k = T(1 - (1 - \frac{k}{N}))$, $k = 1, \dots, N$, une subdivision :

- ▶ si ϕ et φ régulière alors

$$|\langle \mu_T - \hat{\mu}_T, \phi \rangle| \leq CN^{-([\frac{m-1}{2}] \wedge 1/2)} \sup\{\|\nabla_x^j \phi\|_\infty; j \leq m+2\},$$

Erreur (faible) \mathcal{T}

- ▶ **Objectif : contrôler** $|\langle \mu_T - \hat{\mu}_T, \phi \rangle|$
 - ▶ Propriété de Markov : **erreur globale = somme des erreurs locales**
- ▶ **erreur locale** $[T_{k-1}, T_k]$: $E_{T_k} = |\langle \mu_{T_k} - \hat{\mu}_{T_k}, \psi \rangle|$, ψ fonction régulière φ régulière.

Vraie dynamique : $dX_t = V(X_t, \langle \mu_t, \varphi \rangle) \circ dB_t$

Approximation : $d\hat{X}_t^{(j), \hat{\mu}} = V(\hat{X}_t^{(j), \hat{\mu}}, \sum_{p=0}^{q-1} \frac{1}{p!} (t - T_{k-1}) \langle \hat{\mu}_{T_{k-1}}, (\mathcal{L}^{\hat{\mu}})^p \varphi \rangle) dw_t^j$,

$E_{T_k} =$ **erreur Euler** $\approx \Delta_{T_k}^{q+1}$ + propagation d'erreur $\approx \Delta_{T_k} E_{T_{k-1}}$ + erreur de cubature $\approx \Delta_{T_k}^{(m+1)/2}$

Vitesse d'approximation : soit $\mathcal{T}(m, q)$ un arbre et $T_k = T(1 - (1 - \frac{k}{N}))$,
 $k = 1, \dots, N$, une subdivision :

- ▶ si ϕ et φ régulière alors

$$|\langle \mu_T - \hat{\mu}_T, \phi \rangle| \leq CN^{-((m-1)/2) \wedge q} \sup\{\|\nabla_x^j \phi\|_\infty; j \leq m+2\},$$

Erreur (faible) \mathcal{T}

- ▶ **Objectif : contrôler** $|\langle \mu_T - \hat{\mu}_T, \phi \rangle|$
 - ▶ Propriété de Markov : **erreur globale = somme des erreurs locales**
- ▶ **erreur locale** $[T_{k-1}, T_k]$: $E_{T_k} = |\langle \mu_{T_k} - \hat{\mu}_{T_k}, \psi \rangle|$, ψ fonction régulière φ régulière.

Vraie dynamique : $dX_t = V(X_t, \langle \mu_t, \varphi \rangle) \circ dB_t$

Approximation : $d\hat{X}_t^{(j), \hat{\mu}} = V(\hat{X}_t^{(j), \hat{\mu}}, \sum_{p=0}^{q-1} \frac{1}{p!} (t - T_{k-1}) \langle \hat{\mu}_{T_{k-1}}, (\mathcal{L}^{\hat{\mu}})^p \varphi \rangle) dw_t^j$,

$E_{T_k} =$ erreur Euler $\approx \Delta_{T_k}^{q+1}$ + propagation d'erreur $\approx \Delta_{T_k} E_{T_{k-1}}$ + erreur de cubature $\approx \Delta_{T_k}^{(m+1)/2}$

Vitesse d'approximation : soit $\mathcal{T}(m, q)$ un arbre et $T_k = T(1 - (1 - \frac{k}{N}))$,
 $k = 1, \dots, N$, une subdivision :

- ▶ si ϕ et φ régulière alors

$$|\langle \mu_T - \hat{\mu}_T, \phi \rangle| \leq CN^{-((m-1)/2) \wedge q} \sup\{\|\nabla_x^j \phi\|_\infty; j \leq m+2\},$$

si ϕ Lipschitz ?

Erreur (faible) \mathcal{T}

- ▶ **Objectif : contrôler** $|\langle \mu_T - \hat{\mu}_T, \phi \rangle|$
 - ▶ Propriété de Markov : **erreur globale = somme des erreurs locales**
- ▶ **erreur locale** $[T_{k-1}, T_k]$: $E_{T_k} = |\langle \mu_{T_k} - \hat{\mu}_{T_k}, \psi \rangle|$, ψ fonction régulière φ régulière.

Vraie dynamique : $dX_t = V(X_t, \langle \mu_t, \varphi \rangle) \circ dB_t$

Approximation : $d\hat{X}_t^{(j), \hat{\mu}} = V(\hat{X}_t^{(j), \hat{\mu}}, \sum_{p=0}^{q-1} \frac{1}{p!} (t - T_{k-1}) \langle \hat{\mu}_{T_{k-1}}, (\mathcal{L}^{\hat{\mu}})^p \varphi \rangle) dw_t^j$,

$E_{T_k} =$ erreur Euler $\approx \Delta_{T_k}^{q+1}$ + propagation d'erreur $\approx \Delta_{T_k} E_{T_{k-1}}$ + erreur de cubature $\approx \Delta_{T_k}^{(m+1)/2}$

Vitesse d'approximation : soit $\mathcal{T}(m, q, \gamma)$ un arbre et $T_k = T(1 - (1 - \frac{k}{N})^\gamma)$, $k = 1, \dots, N$, une subdivision :

- ▶ if $\gamma = 1$, si ϕ et φ régulière alors

$$|\langle \mu_T - \hat{\mu}_T, \phi \rangle| \leq CN^{-([\frac{m-1}{2}] \wedge q)} \sup\{\|\nabla_x^j \phi\|_\infty; j \leq m+2\},$$

- ▶ **si $\gamma > m - 1$, si ϕ Lipschitz, V uniformément elliptique et φ régulière, alors**

$$|\langle \mu_T - \hat{\mu}_T, \phi \rangle| \leq CN^{-([\frac{m-1}{2}] \wedge q \wedge \gamma/2)} \|\phi\|_{\text{Lip}}.$$

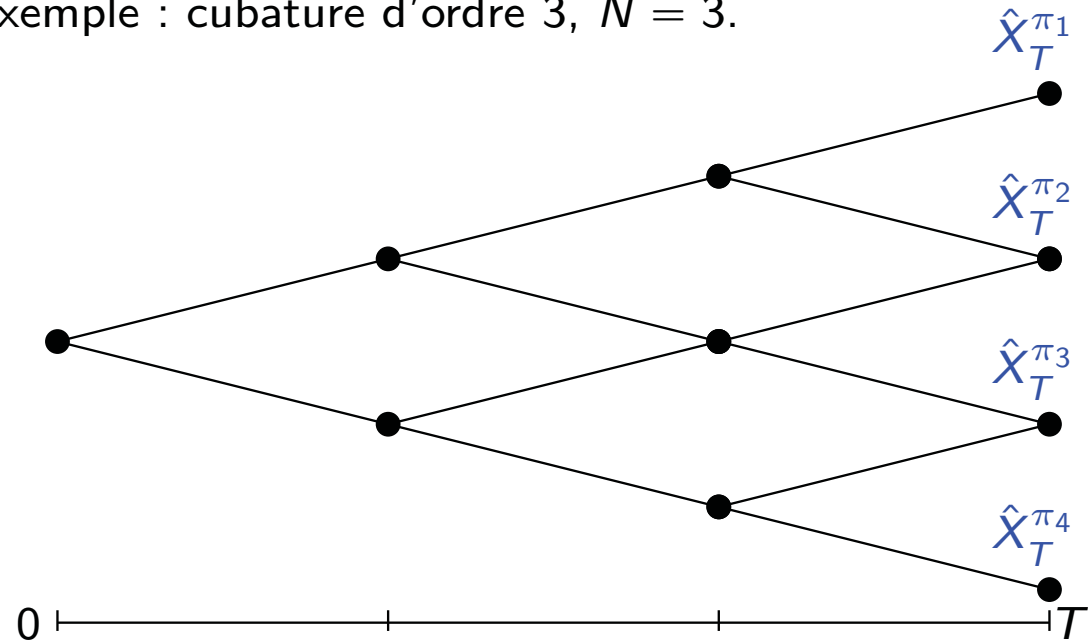
Algorithme, seconde étape : remonter l'arbre $\mathcal{T}(m)$ à rebours

↳ inspiré par Zhang (2004) et Crisan & Manolarakis (2010)

exemple : cubature d'ordre 3, $N = 3$.

Données :

- ▶ intervalle $[0, T]$, $N > 0$
- ▶ $T_0 = 0 < \dots < T_N = T$
- ▶ cubature d'ordre m :
 $\{w_t^1, \dots, w_t^n\}, \{\lambda_1, \dots, \lambda_n\}$
- ▶ arbre $\mathcal{T}(m)$, ϕ
- ▶ $\pi_j \in \{1, \dots, n\}^N$ (généalogie)



Algorithme, seconde étape : remonter l'arbre $\mathcal{T}(m)$ à rebours

↳ inspiré par Zhang (2004) et Crisan & Manolarakis (2010)

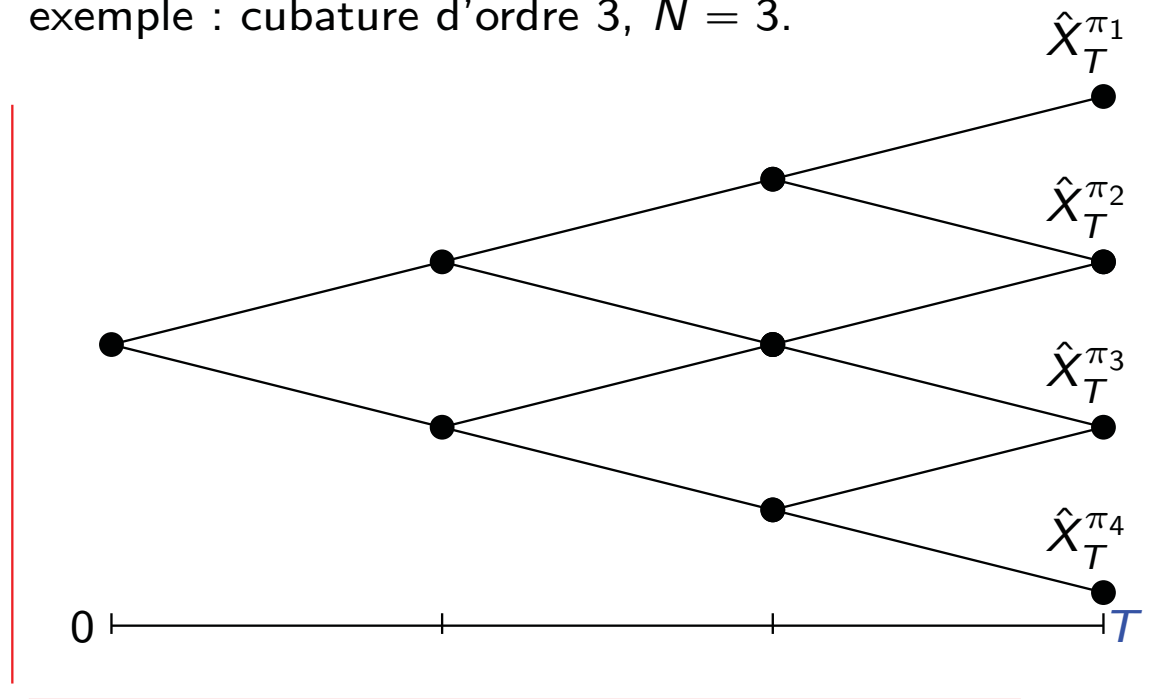
exemple : cubature d'ordre 3, $N = 3$.

Initialisation :

- ▶ intervalle $[0, T]$, $N > 0$
- ▶ $T_0 = 0 < \dots < T_N = T$
- ▶ cubature d'ordre m :

$$\{w_t^1, \dots, w_t^n\}, \{\lambda_1, \dots, \lambda_n\}$$

- ▶ arbre $\mathcal{T}(m)$, ϕ
- ▶ $\pi_j \in \{1, \dots, n\}^N$ (généalogie)



- ▶ Vraie valeur : $Y_T^{\pi_j} = u(T, \hat{X}_T^{\pi_j}) = \phi(X_T^{\pi_j})$, $Z_T^{\pi_j} =: v(T, X_T^{\pi_j}) = V(X_T^{\pi_j}, \mu_T) D_x u(T, X_T^{\pi_j})$

Algorithme, seconde étape : remonter l'arbre $\mathcal{T}(m)$ à rebours

↳ inspiré par Zhang (2004) et Crisan & Manolarakis (2010)

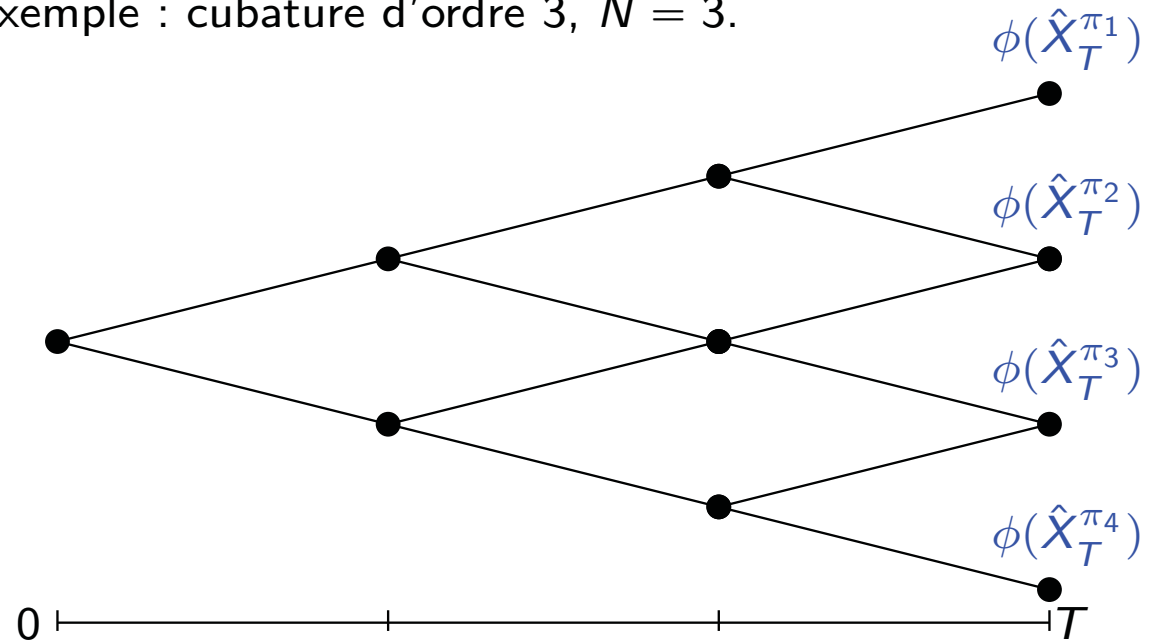
exemple : cubature d'ordre 3, $N = 3$.

Initialisation :

- ▶ intervalle $[0, T]$, $N > 0$
- ▶ $T_0 = 0 < \dots < T_N = T$
- ▶ cubature d'ordre m :

$$\{w_t^1, \dots, w_t^n\}, \{\lambda_1, \dots, \lambda_n\}$$

- ▶ arbre $\mathcal{T}(m)$, ϕ
- ▶ $\pi_j \in \{1, \dots, n\}^N$ (généalogie)



- ▶ Vraie valeur : $Y_T^{\pi_j} = u(T, \hat{X}_T^\pi) = \phi(X_T^{\pi_j})$, $Z_T^{\pi_j} =: v(T, X_T^{\pi_j}) = V(X_T^{\pi_j}, \mu_T) D_x u(T, X_T^{\pi_j})$
- ▶ Approximation : $\hat{Y}_T^\pi = \hat{u}(T, \hat{X}_T^\pi) = \phi(\hat{X}_T^\pi)$, $\hat{Z}_T^\pi = \hat{v}(T, \hat{X}_T^\pi) = 0$

Algorithme, seconde étape : remonter l'arbre $\mathcal{T}(m)$ à rebours

↳ inspiré par Zhang (2004) et Crisan & Manolarakis (2010)

exemple : cubature d'ordre 3, $N = 3$.

étape $[T_{N-1}, T_N[$

▶ intervalle $[0, T]$, $N > 0$

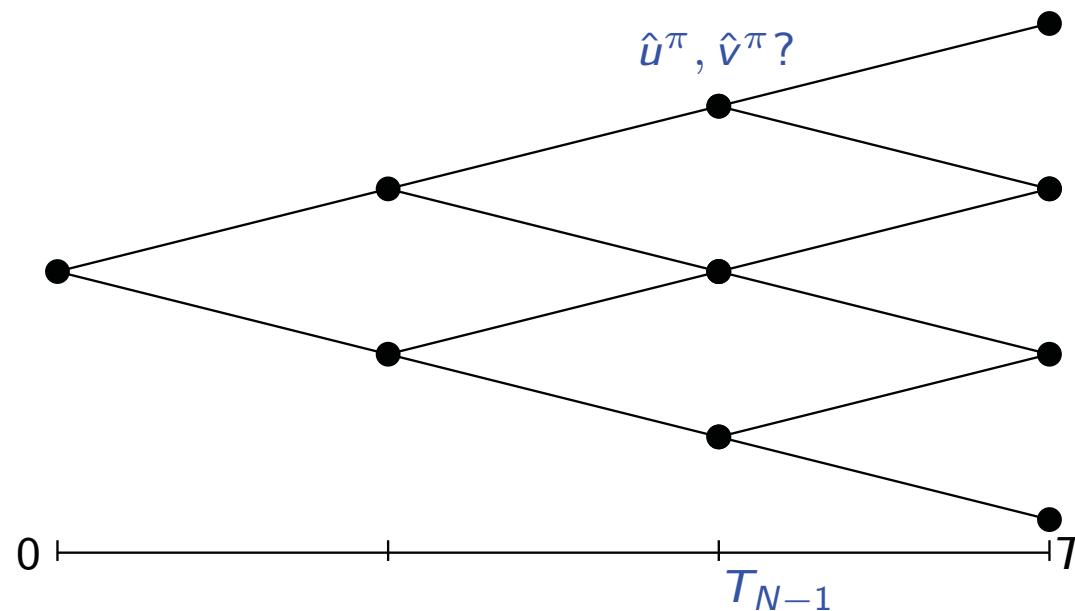
▶ $T_0 = 0 < \dots < T_N = T$

▶ cubature d'ordre m :

$$\{w_t^1, \dots, w_t^n\}, \{\lambda_1, \dots, \lambda_n\}$$

▶ arbre $\mathcal{T}(m)$, ϕ

▶ $\pi \in \{1, \dots, n\}^{N-1}$ (généalogie)



▶ Vraie valeur : $Y_{T_{N-1}}^\pi = u(T_{N-1}, X_{T_{N-1}}^\pi) = \mathbb{E} \left[u(T_N, X_{T_N}^\pi) + \int_{T_{N-1}}^{T_N} f(\dots) ds \mid \mathcal{F}_{T_{N-1}} \right]$

▶ Approximation : $\hat{Y}_{T_{N-1}}^\pi = \hat{u}^\pi$, $\hat{Z}_{T_{N-1}}^\pi = \hat{v}^\pi$

Algorithme, seconde étape : remonter l'arbre $\mathcal{T}(m)$ à rebours

↳ inspiré par Zhang (2004) et Crisan & Manolarakis (2010)

exemple : cubature d'ordre 3, $N = 3$.

étape $[T_{N-1}, T_N[$

▶ intervalle $[0, T]$, $N > 0$

▶ $T_0 = 0 < \dots < T_N = T$

▶ cubature d'ordre m :

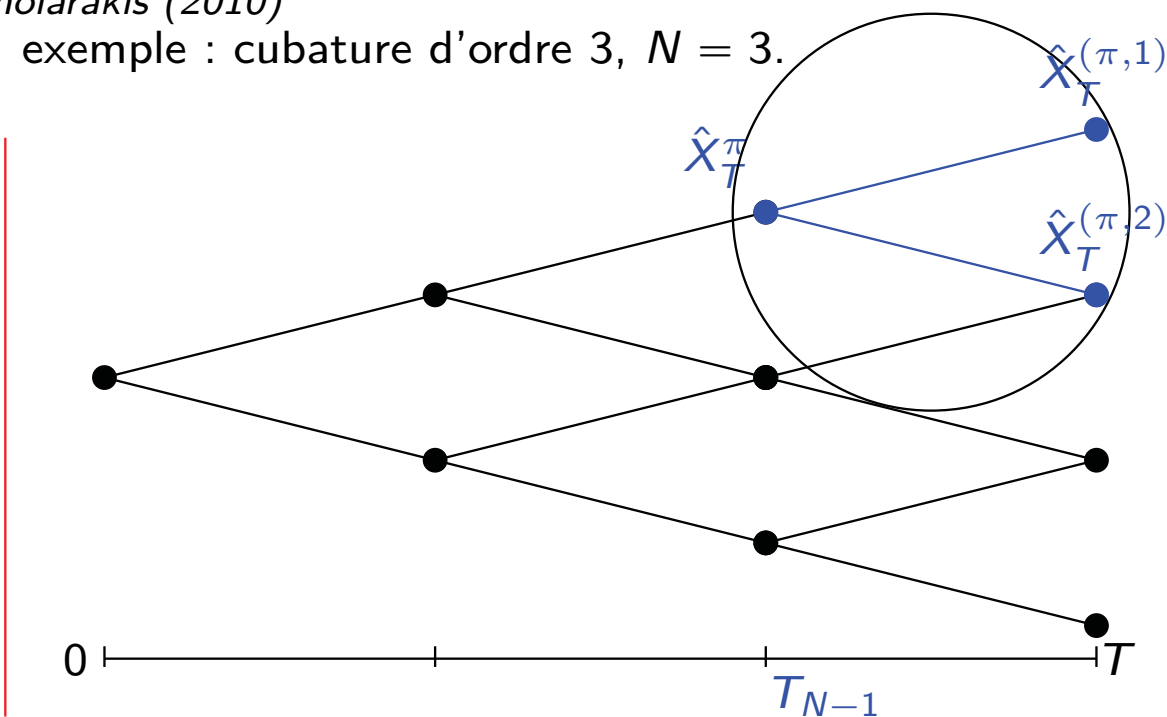
$$\{w_t^1, \dots, w_t^n\}, \{\lambda_1, \dots, \lambda_n\}$$

▶ arbre $\mathcal{T}(m)$, ϕ

▶ $\pi \in \{1, \dots, n\}^{N-1}$ (généalogie)

▶ Vraie valeur : $Y_{T_{N-1}}^\pi = u(T_{N-1}, X_{T_{N-1}}^\pi) = \mathbb{E} \left[u(T_N, X_{T_N}^\pi) + \int_{T_{N-1}}^{T_N} f(\dots) ds \mid \mathcal{F}_{T_{N-1}} \right]$

↳ l'arbre local de cubature donne une approximation de la loi conditionnelle



Algorithme, seconde étape : remonter l'arbre $\mathcal{T}(m)$ à rebours

↳ inspiré par Zhang (2004) et Crisan & Manolarakis (2010)

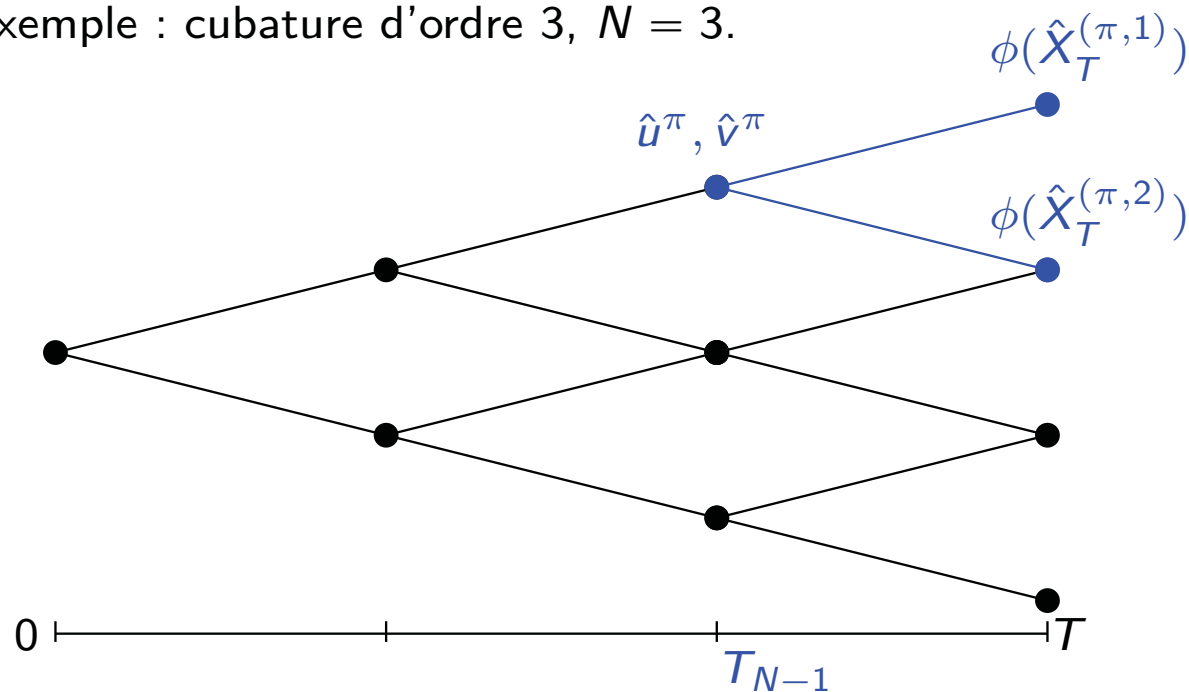
exemple : cubature d'ordre 3, $N = 3$.

étape $[T_{N-1}, T_N[$

- ▶ intervalle $[0, T]$, $N > 0$
- ▶ $T_0 = 0 < \dots < T_N = T$
- ▶ cubature d'ordre m :

$$\{w_t^1, \dots, w_t^n\}, \{\lambda_1, \dots, \lambda_n\}$$

- ▶ arbre $\mathcal{T}(m)$, ϕ
- ▶ $\pi \in \{1, \dots, n\}^{N-1}$ (généalogie)



- ▶ Vraie valeur : $Y_{T_{N-1}}^\pi = u(T_{N-1}, X_{T_{N-1}}^\pi) = \mathbb{E} \left[u(T_N, X_{T_N}^\pi) + \int_{T_{N-1}}^{T_N} f(\dots) ds \mid \mathcal{F}_{T_{N-1}} \right]$

↳ l'arbre local de cubature donne une approximation de la loi conditionnelle

- ▶ $v(T_{N-1}, \hat{X}_{T_{N-1}}^\pi) \leftarrow \mathbb{E} \left[u \left(T_N, X^{T_{N-1}, \hat{X}_{T_N}^\pi} \right) \Delta W_{T_N} \right] \leftarrow \sum_{j=1}^n \lambda_j \phi(\hat{X}_{T_N}^{(\pi,j)}) \Delta w_{T_N}^{(j)}$

$$u(T_{N-1}, \hat{X}_{T_{N-1}}^\pi) \leftarrow \mathbb{E} \left[u(T_{k+1}, X_{T_{k+1}}^{T_k, y}) + \Delta_{T_{k+1}} f(\cdot, \hat{\mu}_{T_{N-1}}) \right] \leftarrow \sum_{j=1}^n \lambda_j \phi(\hat{X}_{T_N}^{(\pi,j)}) + \Delta_{T_N} f(\cdot, \hat{\mu}_{T_{N-1}})$$

Algorithme, seconde étape : remonter l'arbre $\mathcal{T}(m)$ à rebours

↳ inspiré par Zhang (2004) et Crisan & Manolarakis (2010)

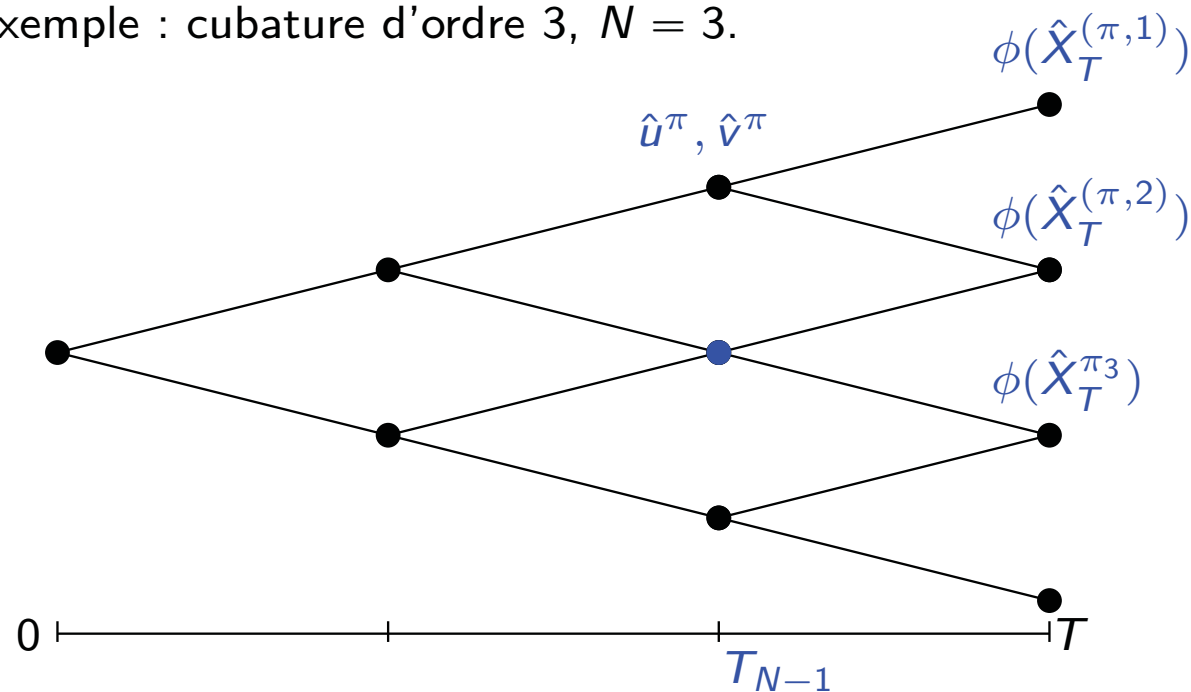
exemple : cubature d'ordre 3, $N = 3$.

étape $[T_{N-1}, T_N[$

- ▶ intervalle $[0, T]$, $N > 0$
- ▶ $T_0 = 0 < \dots < T_N = T$
- ▶ cubature d'ordre m :

$$\{w_t^1, \dots, w_t^n\}, \{\lambda_1, \dots, \lambda_n\}$$

- ▶ arbre $\mathcal{T}(m)$, ϕ
- ▶ $\pi \in \{1, \dots, n\}^{N-1}$ (généalogie)



- ▶ Vraie valeur : $Y_{T_{N-1}}^\pi = u(T_{N-1}, X_{T_{N-1}}^\pi) = \mathbb{E} \left[u(T_N, X_{T_N}^\pi) + \int_{T_{N-1}}^{T_N} f(\dots) ds \mid \mathcal{F}_{T_{N-1}} \right]$

↳ l'arbre local de cubature donne une approximation de la loi conditionnelle

- ▶ $v(T_{N-1}, \hat{X}_{T_{N-1}}^\pi) \leftarrow \mathbb{E} \left[u \left(T_N, X^{T_{N-1}, \hat{X}_{T_N}^\pi} \right) \Delta W_{T_N} \right] \leftarrow \sum_{j=1}^n \lambda_j \phi(\hat{X}_{T_N}^{(\pi, j)}) \Delta w_{T_N}^{(j)}$

$$u(T_{N-1}, \hat{X}_{T_{N-1}}^\pi) \leftarrow \mathbb{E} \left[u(T_{k+1}, X_{T_{k+1}}^{T_k, y}) + \Delta_{T_{k+1}} f(\cdot, \hat{\mu}_{T_{N-1}}) \right] \leftarrow \sum_{j=1}^n \lambda_j \phi(\hat{X}_{T_N}^{(\pi, j)}) + \Delta_{T_N} f(\cdot, \hat{\mu}_{T_{N-1}})$$

Algorithme, seconde étape : remonter l'arbre $\mathcal{T}(m)$ à rebours

↳ inspiré par Zhang (2004) et Crisan & Manolarakis (2010)

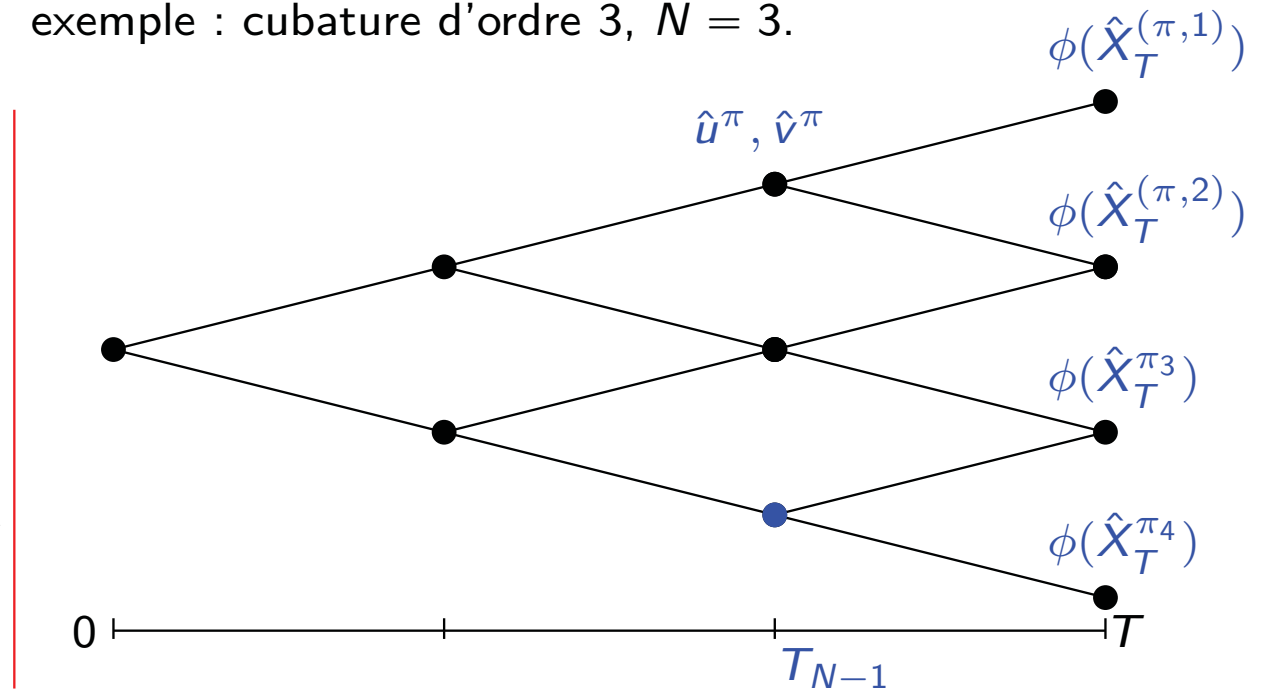
exemple : cubature d'ordre 3, $N = 3$.

étape $[T_{N-1}, T_N[$

- ▶ intervalle $[0, T]$, $N > 0$
- ▶ $T_0 = 0 < \dots < T_N = T$
- ▶ cubature d'ordre m :

$$\{w_t^1, \dots, w_t^n\}, \{\lambda_1, \dots, \lambda_n\}$$

- ▶ arbre $\mathcal{T}(m)$, ϕ
- ▶ $\pi \in \{1, \dots, n\}^{N-1}$ (généalogie)



- ▶ Vraie valeur : $Y_{T_{N-1}}^\pi = u(T_{N-1}, X_{T_{N-1}}^\pi) = \mathbb{E} \left[u(T_N, X_{T_N}^\pi) + \int_{T_{N-1}}^{T_N} f(\dots) ds \mid \mathcal{F}_{T_{N-1}} \right]$

↳ l'arbre local de cubature donne une approximation de la loi conditionnelle

- ▶ $v(T_{N-1}, \hat{X}_{T_{N-1}}^\pi) \leftarrow \mathbb{E} \left[u \left(T_N, X^{T_{N-1}, \hat{X}_{T_N}^\pi} \right) \Delta W_{T_N} \right] \leftarrow \sum_{j=1}^n \lambda_j \phi(\hat{X}_{T_N}^{(\pi,j)}) \Delta w_{T_N}^{(j)}$

$$u(T_{N-1}, \hat{X}_{T_{N-1}}^\pi) \leftarrow \mathbb{E} \left[u(T_{k+1}, X_{T_{k+1}}^{T_k, y}) + \Delta_{T_{k+1}} f(\cdot, \hat{\mu}_{T_{N-1}}) \right] \leftarrow \sum_{j=1}^n \lambda_j \phi(\hat{X}_{T_N}^{(\pi,j)}) + \Delta_{T_N} f(\cdot, \hat{\mu}_{T_{N-1}})$$

Algorithme, seconde étape : remonter l'arbre $\mathcal{T}(m)$ à rebours

↳ inspiré par Zhang (2004) et Crisan & Manolarakis (2010)

exemple : cubature d'ordre 3, $N = 3$.

étape $[T_k, T_{k+1}[$

- ▶ intervalle $[0, T]$, $N > 0$
- ▶ $T_0 = 0 < \dots < T_N = T$
- ▶ cubature d'ordre m :

$$\{w_t^1, \dots, w_t^n\}, \{\lambda_1, \dots, \lambda_n\}$$

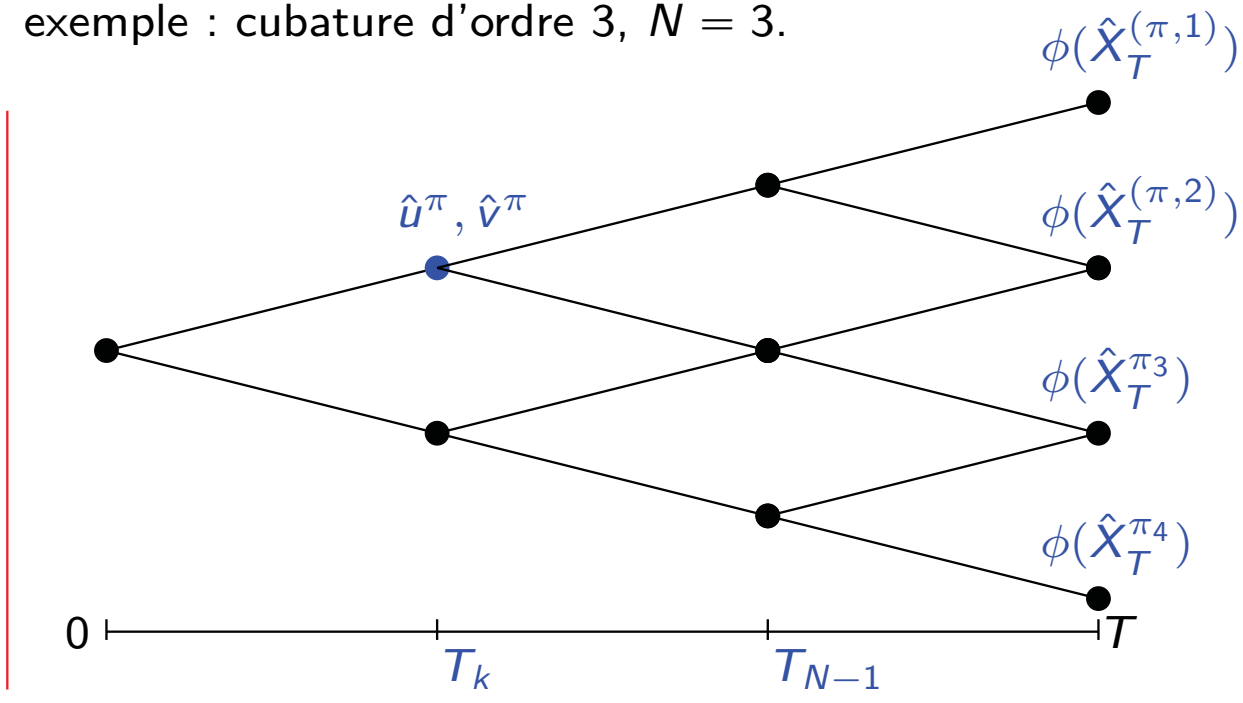
- ▶ arbre $\mathcal{T}(m)$, ϕ
- ▶ $\pi \in \{1, \dots, n\}^k$ (généalogie)

- ▶ Vraie valeur : $Y_{T_k}^\pi = u(T_k, \hat{X}_{T_k}^\pi)$, $Z_{T_k}^\pi =: v(T_k, X_{T_k}^{\pi_j})$
↳ l'arbre local de cubature donne une approximation de la loi conditionnelle

- ▶ Approximation :

$$\hat{v}(T_k, \hat{X}_{T_k}^\pi) = \sum_{j=1}^n \lambda_j u(T_{k+1}, X_{T_{k+1}}^{T_k, \pi, j}) \Delta w_{T_{k+1}}^{(j)}$$

$$\hat{u}(T_k, \hat{X}_{T_k}^\pi) = \sum_{j=1}^n \lambda_j \left(u(T_{k+1}, X_{T_{k+1}}^{T_k, \pi, j}) + \Delta_{T_{k+1}} f(\cdot, \hat{\mu}_{T_k}) \right)$$



Algorithme, seconde étape : remonter l'arbre $\mathcal{T}(m)$ à rebours

↳ inspiré par Zhang (2004) et Crisan & Manolarakis (2010)

exemple : cubature d'ordre 3, $N = 3$.

étape $[T_k, T_{k+1}[$

- ▶ intervalle $[0, T]$, $N > 0$
- ▶ $T_0 = 0 < \dots < T_N = T$
- ▶ cubature d'ordre m :

$$\{w_t^1, \dots, w_t^n\}, \{\lambda_1, \dots, \lambda_n\}$$

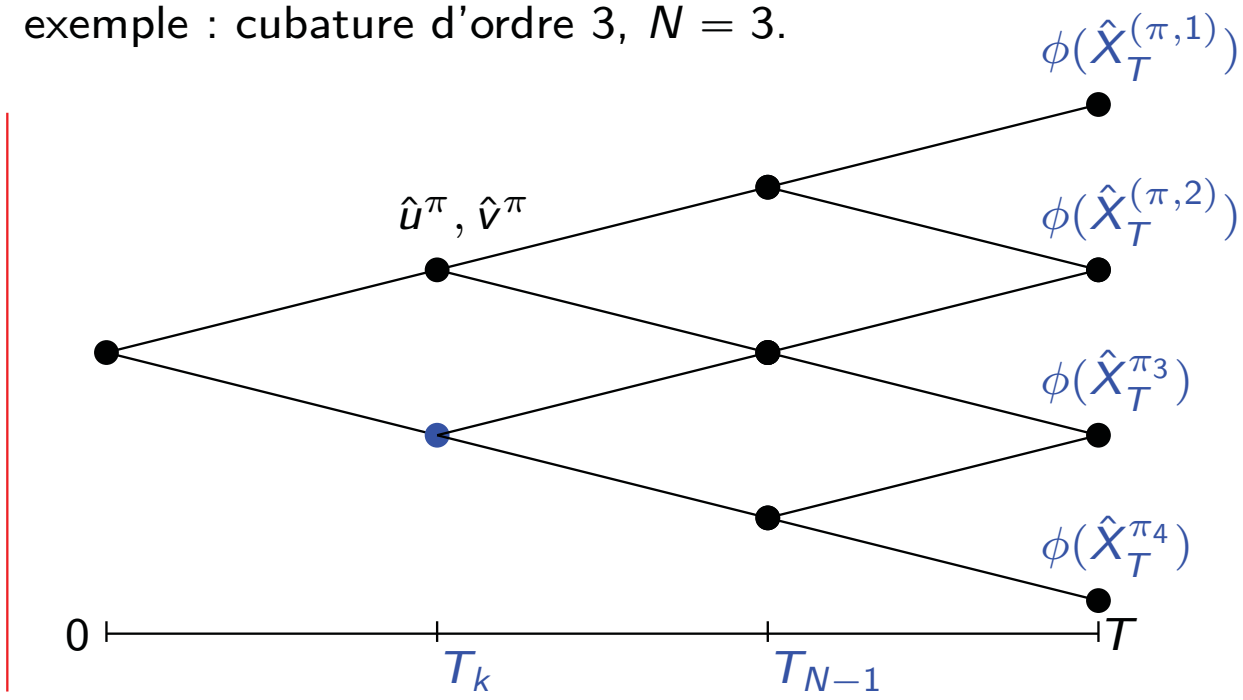
- ▶ arbre $\mathcal{T}(m)$, ϕ
- ▶ $\pi \in \{1, \dots, n\}^k$ (généalogie)

- ▶ Vraie valeur : $Y_{T_k}^\pi = u(T_k, \hat{X}_{T_k}^\pi)$, $Z_{T_k}^\pi =: v(T_k, X_{T_k}^{\pi_j})$
↳ l'arbre local de cubature donne une approximation de la loi conditionnelle

- ▶ Approximation :

$$\hat{v}(T_k, \hat{X}_{T_k}^\pi) = \sum_{j=1}^n \lambda_j u(T_{k+1}, X_{T_{k+1}}^{T_k, \pi, j}) \Delta w_{T_{k+1}}^{(j)}$$

$$\hat{u}(T_k, \hat{X}_{T_k}^\pi) = \sum_{j=1}^n \lambda_j \left(u(T_{k+1}, X_{T_{k+1}}^{T_k, \pi, j}) + \Delta_{T_{k+1}} f(\cdot, \hat{\mu}_{T_k}) \right)$$



Estimation de l'erreur

Soit N un entier, $T_k = T(1 - (1 - \frac{k}{N})^\gamma)$, $k = 1, \dots, N$, une subdivision et $\mathcal{T}(m, q, \gamma)$ un arbre. Sous des hypothèses appropriées on peut montrer qu'il existe une constante C , indépendante de N , telle que

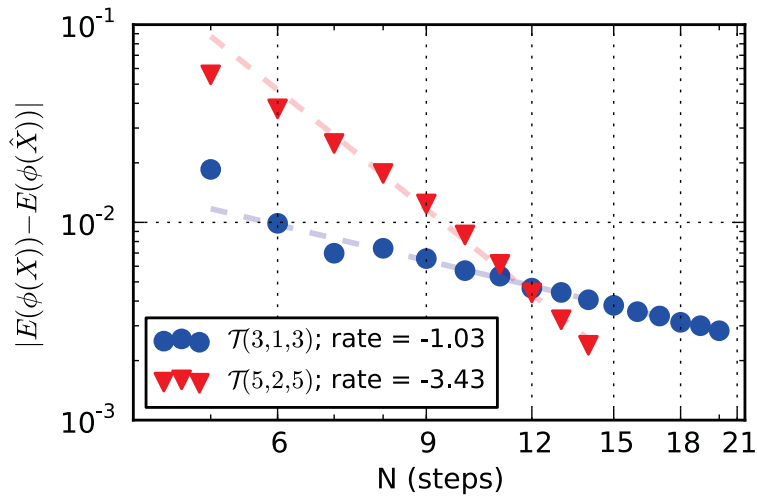
$$\max_{k \in \{0, \dots, N-1\}} \max_{\pi \in \{1, \dots, n\}^k} |\hat{Y}_{T_k}^\pi - Y_{T_k}^\pi| + \Delta_{T_{k+1}}^{1/2} |\hat{Z}_{T_k}^\pi - Z_{T_k}^\pi| \leq C \frac{1}{N}.$$

En outre, on peut construire un schéma de type prédicteur-correcteur pour obtenir l'approximation suivante :

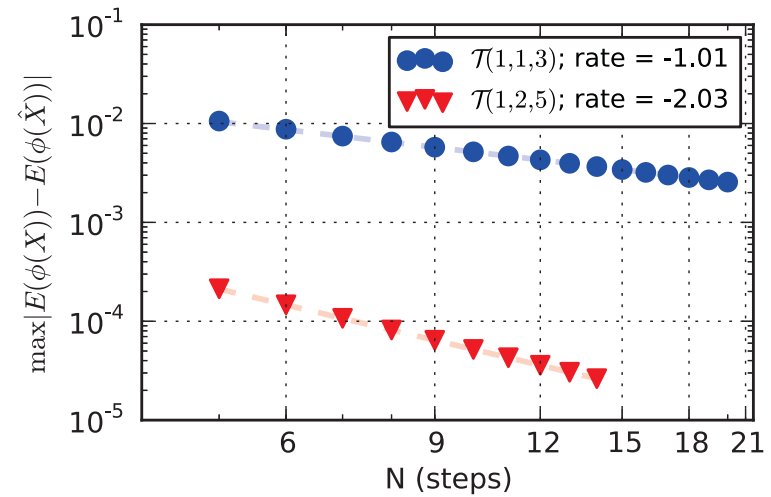
$$\max_{k \in \{0, \dots, N-1\}} \max_{\pi \in \{1, \dots, n\}^k} |\hat{Y}_{T_k}^\pi - Y_{T_k}^\pi| + \Delta_{T_{k+1}}^{1/2} |\hat{Z}_{T_k}^\pi - Z_{T_k}^\pi| \leq C \left(\frac{1}{N}\right)^2.$$

Illustration sur un modèle joué

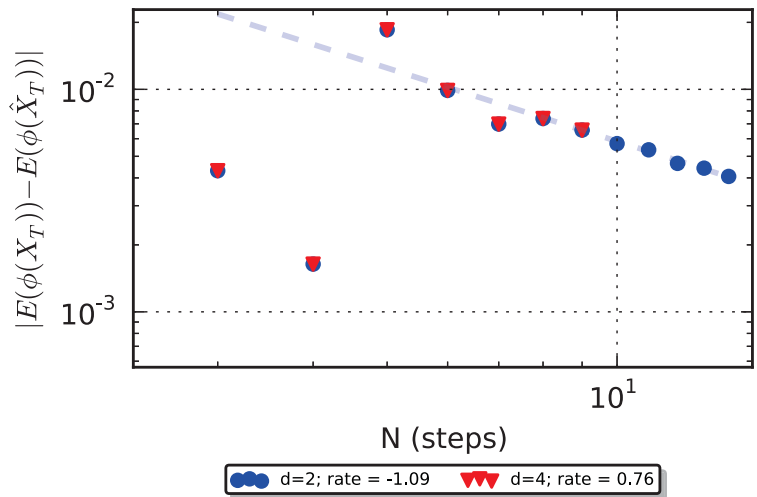
Forward Error - LB



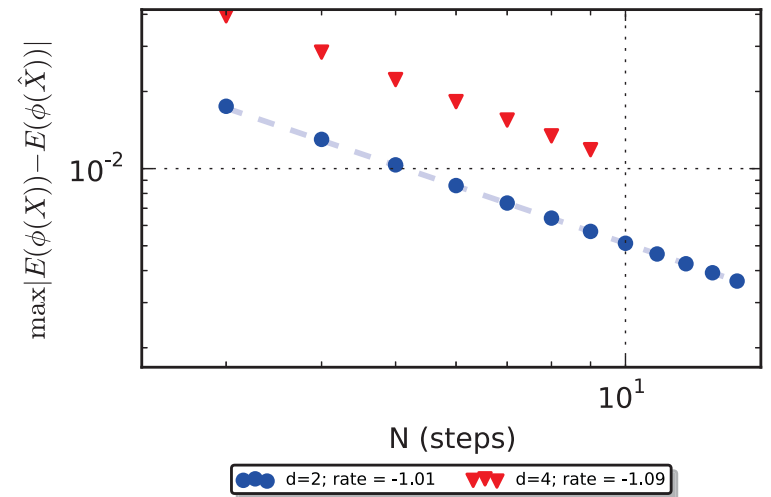
Forward Error - SB



Forward Error - LB



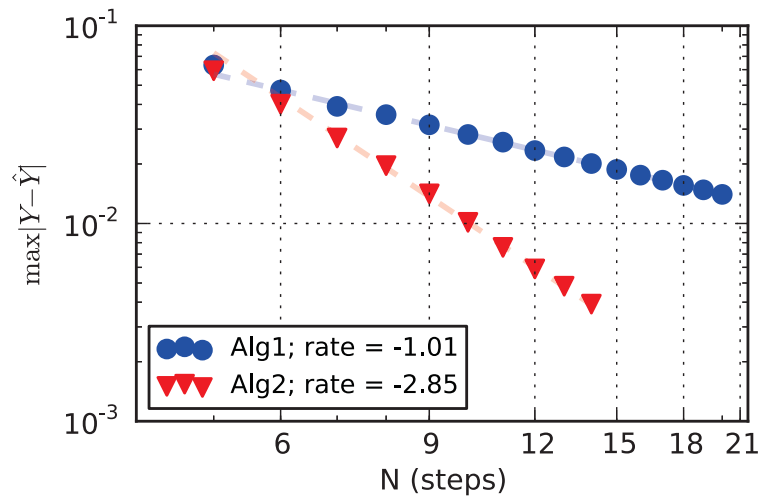
Forward Error - SB



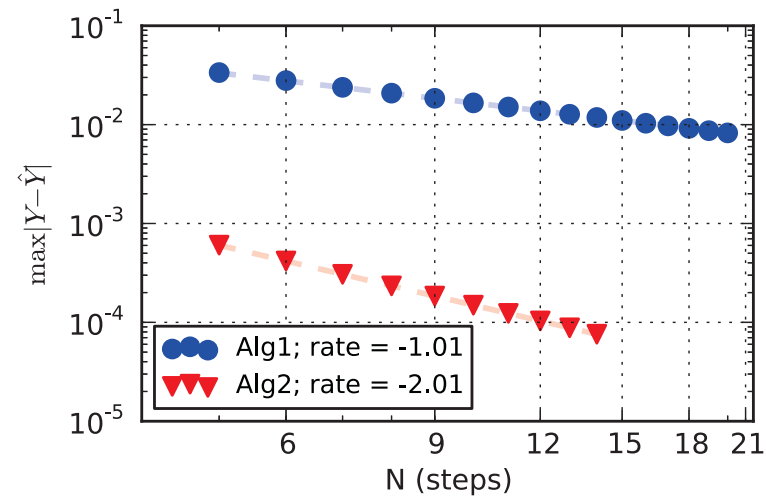
$$dX_t = \mathbb{E}[\sin(X_t)]dt + dB_t; \implies X = B$$

Illustration sur un modèle joué

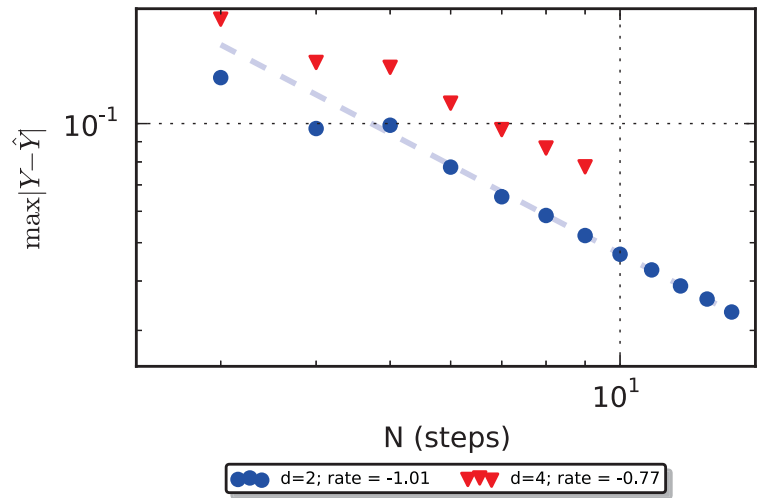
Backward error : Y - LB



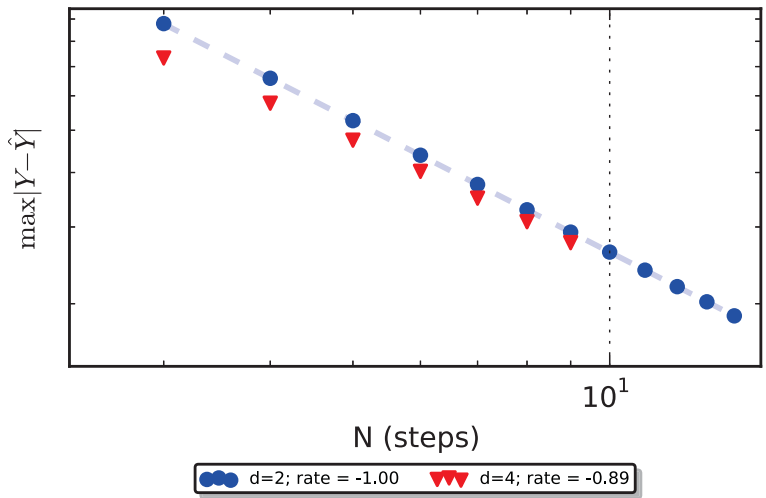
Backward error : Y - SB



Backward error : Y - LB



Backward error : Y - SB



$$dX_t = \mathbb{E}[\sin(X_t)]dt + dB_t; \quad -dY_t = \left(\frac{\mathbf{1} \cdot \cos(X_t)}{2} + \mathbb{E} \left[(\mathbf{1} \cdot \sin(X_t)) \exp(-Y_t^2) \right] \right) dt - Z_t \cdot dB_t,$$

Merci !