RMC voor lineaire ODEs

Isidoor Pinillo Esquivel

Introductie

$$y_t = A(t)y(t). \tag{1}$$

Euler:

(4)

(6)

Introductie

$$y_t = A(t)y(t). \tag{1}$$

Euler:

$$\frac{\tilde{y}(t_k + \Delta t) - \tilde{y}(t_k)}{\Delta t} = \quad A(t_k)\tilde{y}(t_k) \tag{2}$$

(4)

(6)

Introductie

$$y_t = A(t)y(t). \tag{1}$$

Euler:

$$\begin{aligned} \frac{\tilde{y}(t_k + \Delta t) - \tilde{y}(t_k)}{\Delta t} &= A(t_k)\tilde{y}(t_k) && (2)\\ \tilde{y}(t_{k+1}) &= (I + \Delta t A(t_k))\tilde{y}(t_k) && (3)\\ t_{k+1} &= t_k + \Delta t && (4)\\ & && (6) \end{aligned}$$

Introductie

$$y_t = A(t)y(t). \tag{1}$$

Euler:

$$\begin{aligned}
\frac{\tilde{y}(t_k + \Delta t) - \tilde{y}(t_k)}{\Delta t} &= A(t_k)\tilde{y}(t_k) && (2)\\
\tilde{y}(t_{k+1}) &= (I + \Delta t A(t_k))\tilde{y}(t_k) && (3)\\
t_{k+1} &= t_k + \Delta t && (4)\\
Y(T_{k+1}) &= (I + \Delta t A(T_{k+1}))Y(T_k) && (5)\\
T_{k+1} &= T_k + \mathrm{Exp}(\Delta t) && (6)
\end{aligned}$$

$$y_t = A(t)y(t). \tag{1}$$

Euler:

$$\begin{aligned}
\frac{\tilde{y}(t_k + \Delta t) - \tilde{y}(t_k)}{\Delta t} &= A(t_k)\tilde{y}(t_k) && (2)\\
\tilde{y}(t_{k+1}) &= (I + \Delta t A(t_k))\tilde{y}(t_k) && (3)\\
t_{k+1} &= t_k + \Delta t && (4)\\
Y(T_{k+1}) &= (I + \Delta t A(T_{k+1}))Y(T_k) && (5)\\
T_{k+1} &= T_k + \mathrm{Exp}(\Delta t) && (6)
\end{aligned}$$

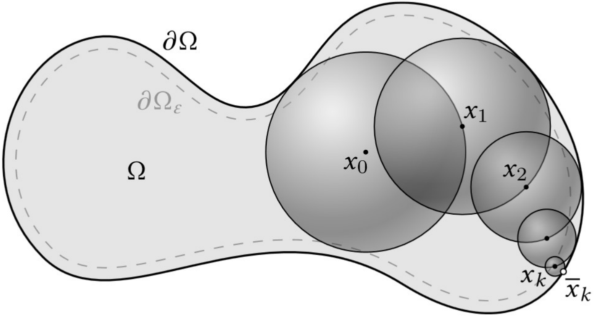
$$y(T_k) = E[Y(T_k)]. \tag{7}$$

Overzicht

1. Introductie
2. Motivatie
3. Monte Carlo
4. Main Poisson algoritme
5. Geavanceerde methoden
6. Conclusie

Motivatie

Veralgemenen van WoS algoritme van (Sawhney e.a. 2022) naar tijd



Monte Carlo

1. Introductie
2. Motivatie
3. Monte Carlo
4. Main Poisson algoritme
5. Geavanceerde methoden
6. Conclusie

Random sample efficientie van optelling

Probleem: benader $\bar{x} = \frac{1}{k}\sum_{i=1}^{k} x_i$ met $n << k$ van x_i's $\in [0, 1]$ (symmetrisch in x_i)

[0] Heinrich en Novak 2001.

Random sample efficientie van optelling

Probleem: benader $\bar{x} = \frac{1}{k}\sum_{i=1}^{k} x_i$ met $n << k$ van x_i's $\in [0,1]$ (symmetrisch in x_i)

Vertrouwensinterval kans $= 1 \sim \frac{k-n}{k} = 1 - \frac{n}{k} \approx 1$ (worst case $\frac{1}{k}$ uit elkaar)

[0] Heinrich en Novak 2001.

Random sample efficientie van optelling

Probleem: benader $\bar{x} = \frac{1}{k}\sum_{i=1}^{k} x_i$ met $n << k$ van x_i's $\in [0,1]$ (symmetrisch in x_i)

Vertrouwensinterval kans $= 1 \sim \frac{k-n}{k} = 1 - \frac{n}{k} \approx 1$ (worst case $\frac{1}{k}$ uit elkaar)

Samplen of termen willekeurig weg laten (Russische roulette)

$$\bar{x} \cong \frac{1}{n}\sum_{i=1}^{n} x_{I_i} \cong \frac{1}{k}\sum_{i=1}^{k} B_i x_i. \tag{8}$$

$\left(E[B_i] = 1, P[B_i = 0] = 1 - \frac{n}{k}\right)$

---

[0] Heinrich en Novak 2001.

Random sample efficientie van optelling

Probleem: benader $\bar{x} = \frac{1}{k}\sum_{i=1}^{k} x_i$ met $n << k$ van x_i's $\in [0,1]$ (symmetrisch in x_i)

Vertrouwensinterval kans $= 1 \sim \frac{k-n}{k} = 1 - \frac{n}{k} \approx 1$ (worst case $\frac{1}{k}$ uit elkaar)

Samplen of termen willekeurig weg laten (Russische roulette)

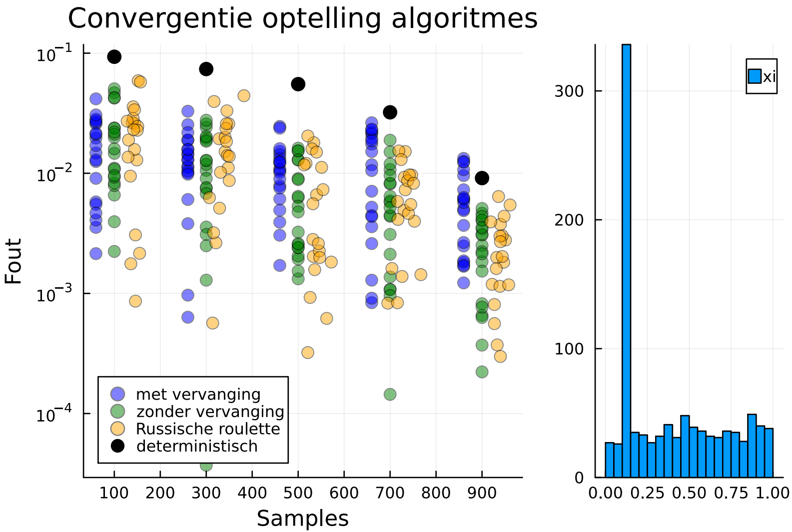
$$\bar{x} \cong \frac{1}{n}\sum_{i=1}^{n} x_{I_i} \cong \frac{1}{k}\sum_{i=1}^{k} B_i x_i. \tag{8}$$

$\left(E[B_i] = 1, P[B_i = 0] = 1 - \frac{n}{k}\right)$

Variantie $=$ RMSE $\sim O\left(\frac{1}{\sqrt{n}}\right)$ en ook vertrouwensintervallen kans < 1 (CLT of Chebychev's ongelijkheid)

[0] Heinrich en Novak 2001.

Optelling algoritmes (plot)



Monte Carlo integratie

Integratie $\approx$ sommatie, integreerbare $f : \mathbb{R} \to [0,1]$:

$$\int_0^1 f(s)ds = E[f(U)] \tag{9}$$

$$\cong \frac{1}{n}\sum_{j=1}^{n} f(U_j) \tag{10}$$

$$\text{met } U_j \sim \text{Uniform}(0,1) \tag{11}$$

1 Introductie

2 Motivatie

3 Monte Carlo

4 Main Poisson algoritme

5 Geavanceerde methoden

6 Conclusie

Main Poisson algoritme

$$y_t = A(t)y, \quad A(t) \text{ meetbaar en } \sup_t ||A(t)|| < \infty \Leftrightarrow \tag{12}$$

$$\tag{15}$$

$$\tag{16}$$

Main Poisson algoritme

$$y_t = A(t)y, \quad A(t) \text{ meetbaar en } \sup_t ||A(t)|| < \infty \Leftrightarrow \tag{12}$$

$$y_t + \sigma y = (\sigma I + A(t))y \Leftrightarrow \tag{13}$$

(15)

(16)

Main Poisson algoritme

$$y_t = A(t)y, \quad A(t) \text{ meetbaar en } \sup_t ||A(t)|| < \infty \Leftrightarrow \tag{12}$$

$$y_t + \sigma y = (\sigma I + A(t))y \Leftrightarrow \tag{13}$$

$$e^{-\sigma t}(e^{\sigma t}y)_t = (\sigma I + A(t))y \Leftrightarrow \tag{14}$$

$$\tag{15}$$

$$\tag{16}$$

Main Poisson algoritme

$$y_t = A(t)y, \quad A(t) \text{ meetbaar en } \sup_t ||A(t)|| < \infty \Leftrightarrow \tag{12}$$

$$y_t + \sigma y = (\sigma I + A(t))y \Leftrightarrow \tag{13}$$

$$e^{-\sigma t}(e^{\sigma t} y)_t = (\sigma I + A(t))y \Leftrightarrow \tag{14}$$

$$y(t) = e^{-\sigma t} y(0) + \int_0^t e^{(s-t)\sigma} \left((\sigma I + A(s))\, y(s)\right) ds, \tag{15}$$

$$\tag{16}$$

Main Poisson algoritme

$$y_t = A(t)y, \quad A(t) \text{ meetbaar en } \sup_t ||A(t)|| < \infty \Leftrightarrow \tag{12}$$

$$y_t + \sigma y = (\sigma I + A(t))y \Leftrightarrow \tag{13}$$

$$e^{-\sigma t}(e^{\sigma t}y)_t = (\sigma I + A(t))y \Leftrightarrow \tag{14}$$

$$y(t) = e^{-\sigma t}y(0) + \int_0^t e^{(s-t)\sigma}\left((\sigma I + A(s))\, y(s)\right) ds, \tag{15}$$

doe volgende substitutie $e^{(s-t)\sigma} = \tau$, equivalent aan exponentieel te samplen

$$\tag{16}$$

Main Poisson algoritme

$$y_t = A(t)y, \quad A(t) \text{ meetbaar en } \sup_t ||A(t)|| < \infty \Leftrightarrow \tag{12}$$

$$y_t + \sigma y = (\sigma I + A(t))y \Leftrightarrow \tag{13}$$

$$e^{-\sigma t}(e^{\sigma t} y)_t = (\sigma I + A(t))y \Leftrightarrow \tag{14}$$

$$y(t) = e^{-\sigma t} y(0) + \int_0^t e^{(s-t)\sigma} \left((\sigma I + A(s)) \, y(s) \right) ds, \tag{15}$$

doe volgende substitutie $e^{(s-t)\sigma} = \tau$, equivalent aan exponentieel te samplen

$$y(t) = \int_0^{e^{-\sigma t}} y(0) d\tau + \int_{e^{-\sigma t}}^1 \left(I + \frac{A(s)}{\sigma} \right) y(s) d\tau. \tag{16}$$

Main Poisson algoritme (recursie)

$$y(t) = \int_0^{e^{-\sigma t}} y(0) d\tau + \int_{e^{-\sigma t}}^{1} \left(I + \frac{A(s)}{\sigma} \right) y(s) d\tau. \tag{17}$$

Monte Carlo integreer,

Main Poisson algoritme (recursie)

$$y(t) = \int_0^{e^{-\sigma t}} y(0)d\tau + \int_{e^{-\sigma t}}^{1} \left(I + \frac{A(s)}{\sigma} \right) y(s)d\tau. \tag{17}$$

Monte Carlo integreer,
doe dit recursief verder:

$$Y(t) = \begin{cases} y(0) & \text{als } e^{-\sigma t} \leq \tau \\ \left(I + \frac{A(S)}{\sigma} \right) Y(S) & \text{anders} \end{cases}, \tag{18}$$

met $S = t + \frac{\ln(\tau)}{\sigma}$.

Main Poisson algoritme (recursie)

$$y(t) = \int_0^{e^{-\sigma t}} y(0) d\tau + \int_{e^{-\sigma t}}^1 \left(I + \frac{A(s)}{\sigma} \right) y(s) d\tau. \tag{17}$$

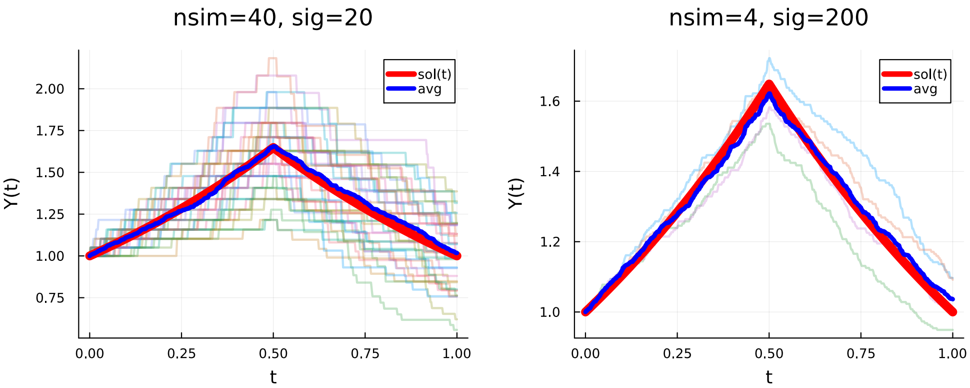
Monte Carlo integreer,
doe dit recursief verder:

$$Y(t) = \begin{cases} y(0) & \text{als } e^{-\sigma t} \leq \tau \\ \left(I + \frac{A(S)}{\sigma} \right) Y(S) & \text{anders} \end{cases}, \tag{18}$$

met $S = t + \frac{\ln(\tau)}{\sigma}$.

$$y(t) = E\left[\prod_{k=0}^{N_t} \left(I + \frac{A(T_k)}{\sigma} \right) \right] y(0). \tag{19}$$

Main Poisson algoritme (convergentie)



Figuur: Realisaties $Y(t)$ met $A(t) = \begin{cases} 1 & \text{voor } t < 0.5, \\ -1 & \text{voor } t \geq 0.5. \end{cases}$ voor verschillende *nsim* en *sig*.

Main Poisson algoritme (opmerkingen)

$$y(t) \cong Y(t) = \prod_{k=0}^{N_t} \left(I + \frac{A(T_k)}{\sigma} \right) y(0). \tag{20}$$

- TB: $E[||Y(t)||], Var[||Y(t)||] < \infty$, wet totale verwachting/variantie
- Parallelle complexiteit
- Inspiratie uit Acebrón en Ribeiro 2016

1 Introductie

2 Motivatie

3 Monte Carlo

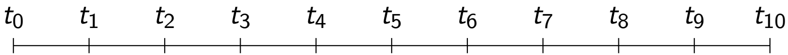
4 Main Poisson algoritme

5 Geavanceerde methoden

6 Conclusie

Recursie in Recursie Monte Carlo

Next flight uit Sawhney e.a. 2022, unbiased methode, 1.5 orde convergentie, moeilijke theorie en te veel functie evaluaties.



$$y_n = \begin{cases} Y_{n-1}(t_n, y_{n-1}) & \text{als } n \neq 0 \\ y(t_0) & \text{als } n = 0 \end{cases} \tag{21}$$

$$Y_n(t, y_n) = y_n + \Delta t B(t) A(S) Y_n(S, y_n) \tag{22}$$

Richting Walk on Spheres

- Plaats discretisatie?
- Walk on components, P transitie matrix, $Py(0)$ met Monte Carlo
- Recursieve formulatie $v^T y(t) = v^T E\left[\prod_{k=0}^{N_t}\left(I + \frac{A(T_k)}{\sigma}\right)\right] y(0)$
- Recursive first passage sampling
- Tijdsafhankelijke Russische roulette voor stijve problemen
- Path stitching vs WoS voor tijd/plaats onafhankelijke problemen

Richting Walk on Spheres

- Plaats discretisatie?
- Walk on components, P transitie matrix, $Py(0)$ met Monte Carlo
- Recursieve formulatie $v^T y(t) = v^T E\left[\prod_{k=0}^{N_t}\left(I + \frac{A(T_k)}{\sigma}\right)\right] y(0)$
- Recursive first passage sampling
- Tijdsafhankelijke Russische roulette voor stijve problemen
- Path stitching vs WoS voor tijd/plaats onafhankelijke problemen

Richting Walk on Spheres

- Plaats discretisatie?
- Walk on components, P transitie matrix, $Py(0)$ met Monte Carlo
- Recursieve formulatie $v^T y(t) = v^T E\left[\prod_{k=0}^{N_t}\left(I + \frac{A(T_k)}{\sigma}\right)\right] y(0)$
- Recursive first passage sampling
- Tijdsafhankelijke Russische roulette voor stijve problemen
- Path stitching vs WoS voor tijd/plaats onafhankelijke problemen

Richting Walk on Spheres

- Plaats discretisatie?
- Walk on components, P transitie matrix, $Py(0)$ met Monte Carlo
- Recursieve formulatie $v^T y(t) = v^T E\left[\prod_{k=0}^{N_t}\left(I + \frac{A(T_k)}{\sigma}\right)\right] y(0)$
- Recursive first passage sampling
- Tijdsafhankelijke Russische roulette voor stijve problemen
- Path stitching vs WoS voor tijd/plaats onafhankelijke problemen

Richting Walk on Spheres

- Plaats discretisatie?
- Walk on components, P transitie matrix, $Py(0)$ met Monte Carlo
- Recursieve formulatie $v^T y(t) = v^T E\left[\prod_{k=0}^{N_t}\left(I + \frac{A(T_k)}{\sigma}\right)\right] y(0)$
- Recursive first passage sampling
- Tijdsafhankelijke Russische roulette voor stijve problemen
- Path stitching vs WoS voor tijd/plaats onafhankelijke problemen

Richting Walk on Spheres

- Plaats discretisatie?
- Walk on components, P transitie matrix, $Py(0)$ met Monte Carlo
- Recursieve formulatie $v^T y(t) = v^T E\left[\prod_{k=0}^{N_t}\left(I + \frac{A(T_k)}{\sigma}\right)\right] y(0)$
- Recursive first passage sampling
- Tijdsafhankelijke Russische roulette voor stijve problemen
- Path stitching vs WoS voor tijd/plaats onafhankelijke problemen

1 Introductie

2 Motivatie

3 Monte Carlo

4 Main Poisson algoritme

5 Geavanceerde methoden

6 Conclusie

Conclusie

- Tevreden met main Poisson algoritme
- Significante stappen in veralgemenen van WoS
- Uitgangspunten voor verder onderzoek

Conclusie

- Tevreden met main Poisson algoritme
- Significante stappen in veralgemenen van WoS
- Uitgangspunten voor verder onderzoek

Conclusie

- Tevreden met main Poisson algoritme
- Significante stappen in veralgemenen van WoS
- Uitgangspunten voor verder onderzoek

Bibliografie I

- Acebrón, Juan A. en Marco A. Ribeiro 2016 "A Monte Carlo method for solving the one-dimensional telegraph equations with boundary conditions".
- Daun, Thomas 2011 "On the randomized solution of initial value problems".
- Ermakov, S. M. en M. G. Smilovitskiy 2021 "The Monte Carlo Method for Solving Large Systems of Linear Ordinary Differential Equations".
- Heinrich, S. en E. Novak 2001 *Optimal Summation and Integration by Deterministic, Randomized, and Quantum Algorithms*.
- Sawhney, Rohan e.a. 2022 "Grid-free Monte Carlo for PDEs with spatially varying coefficients".

Overzicht integratie

Beschouw $f : [0,1]^d \to [0,1]$ k-keer Lipschitz-differentieerbaar:

$$\int_{[0,1]^d} f(x)dx. \tag{23}$$

CVMC $=$ Control Variate MC, $+0.5$ orde over DET algoritmes
FFT $=$ Clenshaw-Curtis/Gauss Quadrature.

	$k=-1$	$0 \leq k \leq 2$	$3 \leq k \leq 5$	$k > 5$
$d = 1$	MC	CVMC	?	FFT
$2 \leq d \leq 5$	MC	CVMC	CVMC	?
$d > 5$	MC	MC	MC	MC

Tabel: Overzicht integratie algoritmes.

Main Poisson appendix

$$y(t) = E\left[\prod_{k=0}^{N_t}\left(I + \frac{A(T_k)}{\sigma}\right)\right] y(0) \tag{24}$$

$$= \sum_{n=0}^{\infty} \frac{\sigma^n t^n}{n!} E\left[\prod_{k=0}^{n}\left(I + \frac{A(T_k)}{\sigma}\right)\right] y(0) \tag{25}$$

Te vergelijken met Magnus expansie.
Voor Monte Carlo transitie matrix hebben we een reductie naar uniformizatie.