Stochastic Computation
– A Brief Introduction –

Klaus Ritter
Computational Stochastics
TU Kaiserslautern
Objectives

Objectives of Computational Stochastics: Construction and analysis of (A) algorithms for computational problems arising in stochastics.
Objectives

Objectives of Computational Stochastics: Construction and analysis of (A) algorithms for computational problems arising in stochastics.

Example: Computation of probabilities, expectations, etc.
in complex stochastic models.

Applications in fluid dynamics, statistical physics, finance, etc.
Objectives

Objectives of Computational Stochastics: Construction and analysis of

(A) algorithms for computational problems arising in stochastics.

(B) stochastic algorithms for computational problems.
Objectives

Objectives of Computational Stochastics: Construction and analysis of

(A) algorithms for computational problems arising in stochastics.

(B) stochastic algorithms for computational problems.

Question: When and how to use a random number generator for
problems from analysis, optimization, stochastics, etc.
Objectives

Objectives of Computational Stochastics: Construction and analysis of

(A) algorithms for computational problems arising in stochastics.

(B) stochastic algorithms for computational problems.

Note: Computational tools vs. computational problems.
Objectives

**Objectives of Computational Stochastics:** Construction and analysis of

(A) algorithms for computational problems arising in stochastics.

(B) stochastic algorithms for computational problems.

**Note:** Computational tools vs. computational problems.

In the sequel, for illustration:

- infinite-dimensional integration for stochastic differential equations.
Stochastic Differential Equations

Consider an autonomous, scalar SDE

\[ dX_t = \mu(X_t) \, dt + \sigma(X_t) \, dW_t, \quad t \in [0, T], \]

\[ X_0 = x_0 \]

with

- a scalar Brownian motion \( W = (W_t)_{t \in [0, T]} \),
- drift and diffusion coefficients \( \mu, \sigma : \mathbb{R} \to \mathbb{R} \),
- and initial value \( x_0 \in \mathbb{R} \).
Consider an autonomous, scalar SDE

\[ dX_t = \mu(X_t) \, dt + \sigma(X_t) \, dW_t, \quad t \in [0, T], \]

\[ X_0 = x_0 \]

with

- a scalar Brownian motion \( W = (W_t)_{t \in [0,T]} \),
- drift and diffusion coefficients \( \mu, \sigma : \mathbb{R} \rightarrow \mathbb{R} \),
- and initial value \( x_0 \in \mathbb{R} \).

Assumption

\[ \mu, \sigma \in \text{Lip}_1(\mathbb{R}, \mathbb{R}). \]
Stochastic Differential Equations

Consider an autonomous, scalar SDE

\[ dX_t = \mu(X_t) \, dt + \sigma(X_t) \, dW_t, \quad t \in [0, T], \]
\[ X_0 = x_0 \]

with

- a scalar Brownian motion \( W = (W_t)_{t \in [0, T]} \),
- drift and diffusion coefficients \( \mu, \sigma : \mathbb{R} \to \mathbb{R} \),
- and initial value \( x_0 \in \mathbb{R} \).

The solution \( X = (X_t)_{t \in [0, T]} \) is a Markov process with continuous paths, and

\[
\lim_{h \to 0} h^{-1} \cdot E(X_{t+h} - X_t \mid X_t = x) = \mu(x),
\]
\[
\lim_{h \to 0} h^{-1/2} \cdot \left( E((X_{t+h} - X_t)^2 \mid X_t = x) \right)^{1/2} = |\sigma|(x).
\]
The Euler Scheme

The Euler scheme with time-discretization

\[ t_k = t_k^K = k/K \cdot T \]

for \( k = 0, \ldots, K \) is defined by

\[
\hat{X}_{t_0} = x_0, \\
\hat{X}_{t_{k+1}} = \hat{X}_{t_k} + \mu(\hat{X}_{t_k}) \cdot (t_{k+1} - t_k) \\
\quad + \sigma(\hat{X}_{t_k}) \cdot (W_{t_{k+1}} - W_{t_k}),
\]

where \( W_{t_{k+1}} - W_{t_k} \) is independent and identically distributed as \( N(0, T/K) \).

and piecewise linear interpolation.

This yields a stochastic process \( \hat{X} = \hat{X}^K = (\hat{X}_t)_{t \in [0,T]} \).
The Computational Problem

Given $x_0 = 1$, $\mu$, $\sigma$, and $f$:

$$f : C([0, T]) \to \mathbb{R},$$

compute

$$S(\mu, \sigma, f) = \mathbb{E}(f(X)) = \int_{C([0, T])} f \, dP_X.$$
The Computational Problem

Given \( x_0 = 1, \mu, \sigma, \) and

\[
 f : C([0, T]) \to \mathbb{R},
\]

compute

\[
 S(\mu, \sigma, f) = \mathbb{E}(f(X)) = \int_{C([0,T])} f \, dP_X.
\]

\( \hookrightarrow \) Infinite-dimensional integration w.r.t. a measure that is given only implicitly.
The Computational Problem

Given $x_0 = 1$, $\mu$, $\sigma$, and

$$f : C([0, T]) \to \mathbb{R},$$

compute

$$S(\mu, \sigma, f) = \mathbb{E}(f(X)) = \int_{C([0,T])} f \, dP_X.$$ 

Assumption: $(\mu, \sigma, f) \in F$ for

$$F = \text{Lip}_1(\mathbb{R}, \mathbb{R}) \times \text{Lip}_1(\mathbb{R}, \mathbb{R}) \times \text{Lip}_1(C([0,T]), \mathbb{R}).$$
The Computational Problem

Given \( x_0 = 1, \mu, \sigma, \) and

\[
f : C([0, T]) \to \mathbb{R},
\]

compute

\[
S(\mu, \sigma, f) = E(f(X)) = \int_{C([0,T])} f \, dP_X.
\]

Monte Carlo Euler, with independent copies \( \hat{X}_K^1, \ldots, \hat{X}_K^M \) of \( \hat{X}^K, \)

\[
S^{K,M}(\mu, \sigma, f) = 1/M \cdot \sum_{m=1}^{M} f(\hat{X}_m^K)
\]
The Computational Problem

Given \( x_0 = 1, \mu, \sigma, \) and

\[
f : C([0, T]) \rightarrow \mathbb{R},
\]

compute

\[
S(\mu, \sigma, f) = \mathbb{E}(f(X)) = \int_{C([0, T])} f \, dP_X.
\]

Monte Carlo Euler, with independent copies \( \hat{X}^K_1, \ldots, \hat{X}^K_M \) of \( \hat{X}^K \),

\[
S^{K,M}(\mu, \sigma, f) = \frac{1}{M} \cdot \sum_{m=1}^{M} f(\hat{X}^K_m)
\]

Questions:

- Quality of \( S^{K,M} \)?
- Are there better algorithms?
Resources

- real number machine; cost one per operation,
- ideal random number generator; cost one per call,
- oracles for $\mu(x)$ and $\sigma(x)$ for any $x \in \mathbb{R}$; cost one per call,
- oracle for $f(x)$ for any piecewise linear function $x \in C([0, T])$ with equidistant breakpoints; cost $= \#\text{breakpoints} + 2$. 
Resources

- real number machine; cost one per operation,
- ideal random number generator; cost one per call,
- oracles for $\mu(x)$ and $\sigma(x)$ for any $x \in \mathbb{R}$; cost one per call,
- oracle for $f(x)$ for any piecewise linear function $x \in C([0, T])$ with equidistant breakpoints; cost $= \#\text{breakpoints} + 2$.

Hereby, we get the definition of a randomized algorithm $\mathcal{A}$ and its cost, $\text{cost}(\mathcal{A}, (\mu, \sigma, f), \omega)$. 
Resources

- real number machine; cost one per operation,
- ideal random number generator; cost one per call,
- oracles for $\mu(x)$ and $\sigma(x)$ for any $x \in \mathbb{R}$; cost one per call,
- oracle for $f(x)$ for any piecewise linear function $x \in C([0, T])$ with equidistant breakpoints; cost $= \# \text{breakpoints} + 2$.

Hereby, we get the definition of a randomized algorithm $\mathcal{A}$ and its cost,

$$\text{cost}(\mathcal{A}, (\mu, \sigma, f), \omega).$$

Maximal error and cost of $\mathcal{A}$

$$\text{error}(\mathcal{A}) = \sup_{(\mu, \sigma, f) \in F} \left( \mathbb{E}(S(\mu, \sigma, f) - \mathcal{A}(\mu, \sigma, f))^2 \right)^{1/2},$$

$$\text{cost}(\mathcal{A}) = \sup_{(\mu, \sigma, f) \in F} \mathbb{E}(\text{cost}(\mathcal{A}, (\mu, \sigma, f), \cdot)).$$
Results

**Thm.** With suitable parameters, Monte Carlo Euler $\mathcal{A}$ achieves

$$\text{error}(\mathcal{A}) \preceq \log \text{cost}(\mathcal{A})^{-1/4}.$$
Results

**Thm.** With suitable parameters, Monte Carlo Euler $\mathcal{A}$ achieves

$$\text{error}(\mathcal{A}) \leq_{\log} \text{cost}(\mathcal{A})^{-1/4}.$$  

**Thm.** With suitable parameters, multi-level Monte Carlo Euler $\mathcal{A}$ achieves

$$\text{error}(\mathcal{A}) \leq_{\log} \text{cost}(\mathcal{A})^{-1/2}.$$  

See Heinrich (1998), Giles (2008), ...
Results

Thm. With suitable parameters, Monte Carlo Euler $\mathcal{A}$ achieves

$$\text{error}(\mathcal{A}) \leq_{\log} \text{cost}(\mathcal{A})^{-1/4}.$$ 

Thm. With suitable parameters, multi-level Monte Carlo Euler $\mathcal{A}$ achieves

$$\text{error}(\mathcal{A}) \leq_{\log} \text{cost}(\mathcal{A})^{-1/2}.$$ 


Thm. For the $n$-th minimal error

$$e(n) = \inf \{\text{error}(\mathcal{A}) : \mathcal{A} \text{ randomized alg. with } \text{cost}(\mathcal{A}) \leq n\}$$

we have

$$e(n) \geq_{\log} n^{-1/2}.$$ 