Moving Particles: the parallel optimal Multilevel Splitting method with applications in quantile estimation and meta-model based algorithms

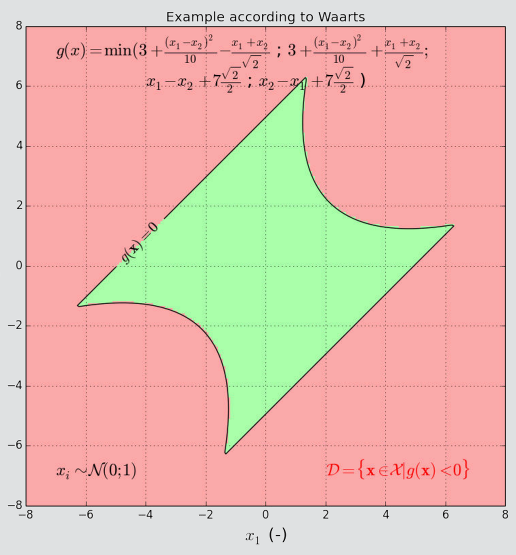
Clément WALTER[1,2], Gilles DEFAUX[1] & Josselin GARNIER[2]

Introduction

- Let $\mathbf{X} \in \mathcal{X} \subseteq \mathbb{R}^d$ be a random vector
- Let $g : \mathbb{R}^d \to \mathbb{R}$ be a measurable function defining the failure domain $D = \{\mathbf{x} \in \mathbb{R}^d \mid g(\mathbf{x}) < q\}$
- Goal: probability measure of D or find q for a given p:

$$p = \mathrm{P}[\mathbf{X} \in D] = \mu^X(D) = \int_D d\mu^X = \int_{\mathbb{R}^d} 1_D(\mathbf{x}) d\mu^X$$

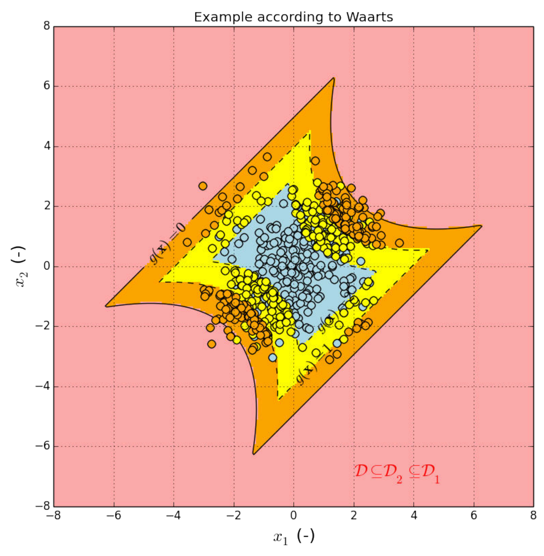
- Constraints:
  - g is a "black-box" whose outputs are time consuming
  - $p \ll 1$, typically $p < 10^{-5}$



State-of-the-art

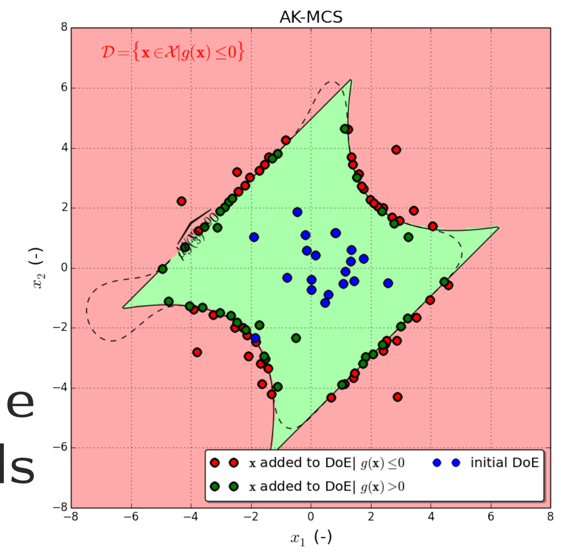
Multilevel Splitting methods (*Subset Simulation*) [1, 2]:

- Write D as $D = \bigcap_{i=1}^{m} D_i = \bigcap_{i=1}^{m} \{g(\mathbf{X}) < q_i\}$ with $(q_i)_i$ a decreasing sequence
- $p = \mathbb{P}[\mathbf{X} \in D] = \prod_i \mathbb{P}[\mathbf{X} \in D_{i+1} \mid \mathbf{X} \in D_i]$
- Primary choice of $(D_i)_i$ or adaptive construction with a cut-off probability p_0
  - optimal value for p_0?
  - adaptive choice brings bias
  - only sequential parallelisation and no quantile estimator



Meta-model based algorithms:

- Spend the computational budget in fitting a surrogate model to g
- Criticality of the Design of Experiments (DoE)
  - density-based DoE are unlucky to produce failing samples
  - uniform DoE depend a lot on the dimension



$\Longrightarrow$ both strategies suffer indeed from the same difficulty: *getting into* extreme levels of g

Move of particles

Let F be the *cdf* of $g(\mathbf{X})$ (assumed to be continuous) and $\Lambda(y) = -\log(F(y))$

Move of one particle along the levels of g [3]

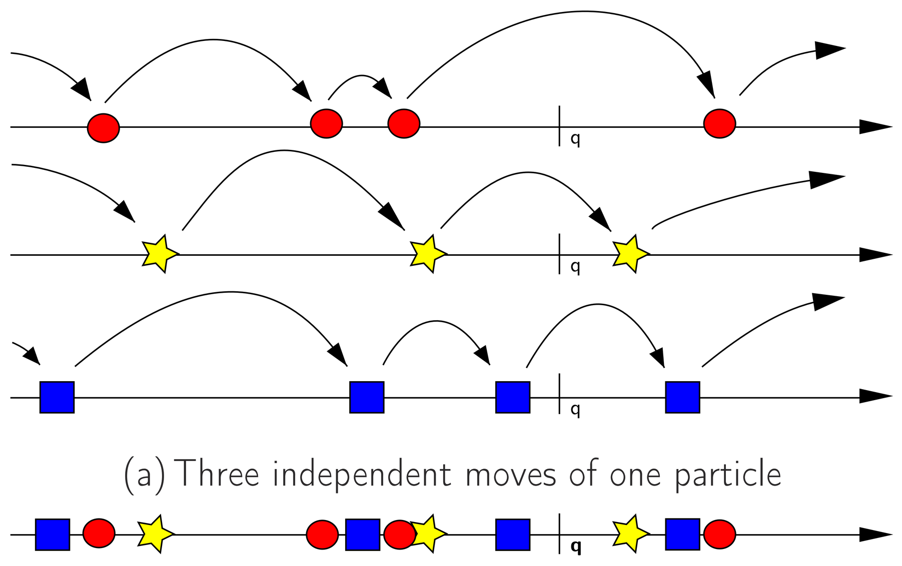
- $q_0 = +\infty$
- For all $m \in \mathbb{N}$:
  - Sample $\mathbf{X} \sim \mu^X(\cdot \mid g < q_m)$
  - Evaluate g: $q_{m+1} = g(\mathbf{X})$

The random variables $(T_m)_{m\geq 1} = (\Lambda(q_m))_{m \geq 1}$ *are distributed as the successive arrival times of a* ***Poisson Process with parameter 1***

Move of N particles along the levels of g [4, 3]

- $\mathbf{q}_0 = (q_0^1, \cdots, q_0^N) = (+\infty, \cdots, +\infty)$
- For all $m \in \mathbb{N}$
  - $\mathbf{q}_{m+1} = \mathbf{q}_m$
  - $i_m = \operatorname{argmax}_i (q_m^i)$
  - Sample $\mathbf{X}_{i_m} \sim \mu^X(\cdot \mid g < q_m^{i_m})$
  - Evaluate g: $q_{m+1}^{i_m} = g(\mathbf{X}_{i_m})$

The RV $(T_m)_{m \geq N} = (\Lambda(q_m^{i_m}))_{m \geq N}$ *are distributed as the successive arrival times of a* ***marked Poisson Process with parameter N***



(a) Three independent moves of one particle

(b) Marked times of Poisson Process with parameter 3

Figure: Moves of particles seen as a realisation of a marked Poisson process

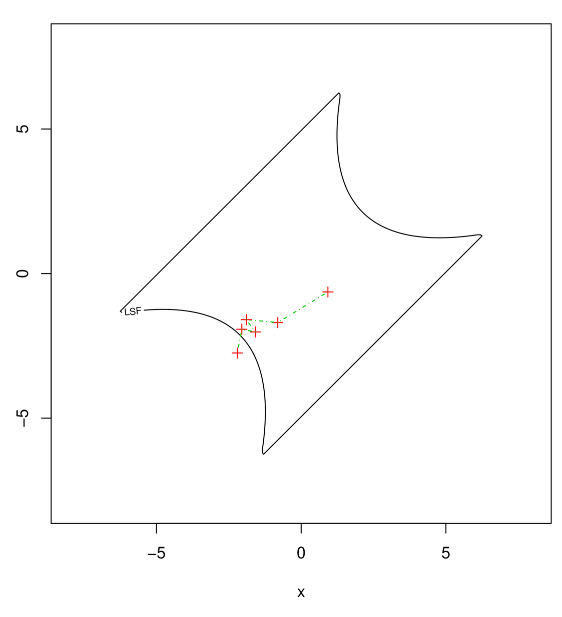


Figure: Example of a move of a particle to the failure domain

Main result: it requires *only* $\mathcal{P}(\log 1/p)$ samples to get a realisation of $\mathbf{X}$ in a domain of measure p while usual sampling needs $\approx 1/p$ samples.

Practical implementation

- Conditional laws $\Rightarrow$ Metropolis-Hastings algorithm
- Insure independence $\Rightarrow$ *burn-in* parameter T
- Almost fully parallel: $N \geq 10$ enough for conditional sampling

Application in rare event probability estimation

- $\Lambda(q) = -\log p$: the number of moves M_t to get into the failure domain is the number of events before time $t = -\log p$: $M_t \sim \mathcal{P}(-N \log p)$
- **Estimate a probability through an estimation of a Poisson parameter** λ given K realisations $(M_k)_k$ [3]:

$$\widehat{\lambda} = \frac{1}{K} \sum_k M_k$$

- With $\lambda = -N \log p$, we look indeed for $\exp(-\lambda/N)$

$$\widehat{p} = \exp(-1/KN)^{\sum M_k} \longrightarrow \widehat{p} = \left(1 - \frac{1}{KN}\right)^{\sum M_k}$$

to get **an unbiased estimator** [4] almost achieving Cramer-Rao bound

- $\mathrm{E}[\widehat{p}] = p$; $\mathrm{CV}[\widehat{p}] = p^{-1/KN} - 1$
- Comparison for a given targeted CV δ and number of cores n_c

Method	Eff. comp. time
Naive MC	$\dfrac{1}{p n_c \delta^2}$
MS ([1, 2])	$\dfrac{T(\log p)^2}{n_c \delta^2} \dfrac{(1-p_0)^2}{p_0 (\log p_0)^2}$
MP [3]	$\dfrac{T(\log p)^2}{n_c \delta^2}$

Table: Moving particles VS usual strategies

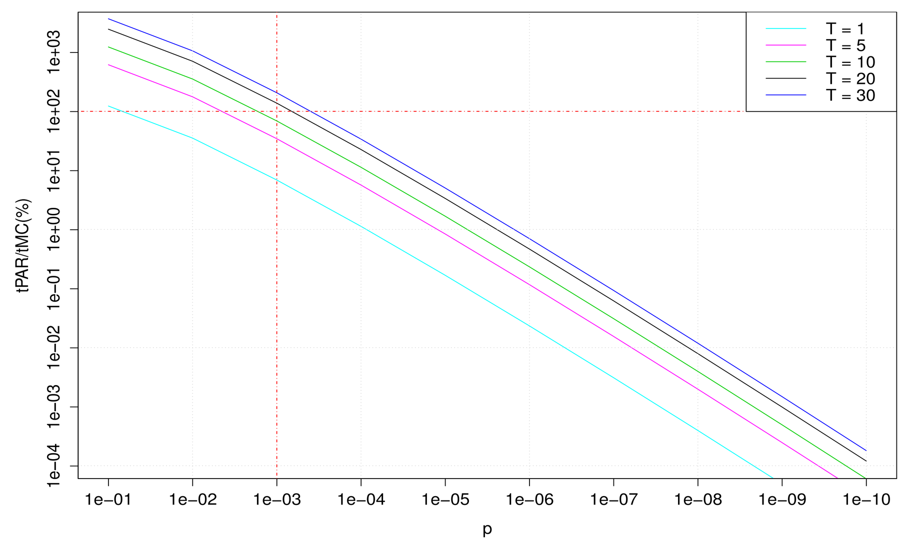


Figure: Eff. computing time MP VS naive MC

Application in quantiles estimation

- **Estimate "time"** $q = \Lambda^{-1}(p)$ **with the realisation of a point process**
- The random counting variable at "time" q: $M_q \sim \mathcal{P}(-N \log p)$
- Time is unknown $\rightarrow$ impossible to get realisations of M_q
- Move a total of N particles; $m = \lceil -N \log p \rceil$ and calculate [3]:

$$\widehat{q} = \frac{1}{2}(q_{m-1} + q_m)$$

- Central Limit Theorem:

$$\sqrt{N}(\widehat{q} - q) \xrightarrow[N \to \infty]{\mathcal{L}} \mathcal{N}\left(0, \frac{-p^2 \log p}{f(q)^2}\right)$$

- Bounds on bias on $1/N$, centred with exponential tails
- Confidence interval available **without estimation of the density at quantile** q
- Computing times similar to those obtained for probability estimator

Application in getting a first DoE

- Number of samples to get one realisation into $D \approx -\log p$
- Metropolis-Hastings increases calls to g to insure convergence but not necessary if one only intends to *move*
- Learn a meta-model on-the-fly while sampling to the failure domain and use it instead of g for the conditional sampling

$$N_{\mathrm{DoE}} \approx d + 1 + N_{\mathrm{failing}} \times \log 1/p$$

Conclusion

- Moving Particles point of view leads to the parallelisation of the optimal Multilevel Splitting method [4], **resolving the issue of choosing the sequence** $(D_i)_i$ **or the cut-off probability** p_0
- The estimator is **unbiased** with lowest variance and **1.5x faster than usual Subset Simulation [1] with** $p_0 = 0.1$
- MP point of view also provides an **optimal parallel quantile estimator**
- It allows also for **quick/cheap access to the failure domain** for surrogate based algorithms

References

[1] S.-K. Au and J. L. Beck *Probabilistic Engineering Mechanics*, vol. 16, no. 4, pp. 263–277, 2001.

[2] F. Cérou, P. Del Moral, T. Furon, and A. Guyader *Statistics and Computing*, vol. 22, no. 3, pp. 795–808, 2012.

[3] C. Walter and ... *To be published*.

[4] A. Guyader, N. Hengartner, and E. Matzner-Løber *Applied Mathematics & Optimization*, vol. 64, no. 2, pp. 171–196, 2011.

[1] CEA, DAM, DIF, F-91297 Arpajon, France; [2] Université Paris Diderot - Paris 7, Paris, France

clement.walter@cea.fr