



HAL
open science

DRlib: a C++ library for marked Gibbs point processes simulation and inference

Didier Gemmerlé, Radu S Stoica, Christophe Reype

► To cite this version:

Didier Gemmerlé, Radu S Stoica, Christophe Reype. DRlib: a C++ library for marked Gibbs point processes simulation and inference. 21st Annual Conference of the International Association for Mathematical Geosciences, IAMG 2022, Aug 2022, Nancy, France. . hal-04047676

HAL Id: hal-04047676

<https://hal.science/hal-04047676>

Submitted on 27 Mar 2023

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

DRlib: a C++ library for marked Gibbs point processes simulation and inference

Didier Gemmerlé¹, Radu S. Stoica², Christophe Reype²

¹ Université de Lorraine, CNRS, IECL, F-54000 Nancy, France

² Université de Lorraine, CNRS, IECL, Inria, F-54000 Nancy, France

{didier.gemmerle,radu-stefan.stoica,christophe.reype}@univ-lorraine.fr

Purpose

DRlib is a new **C++** library for modeling, simulation and inference of marked point processes. The aim is to complete existing tools such as the **spatstat** library in **R** [1] with reliable and efficient **C++** code allowing intensive Bayesian MCMC based inference for marked Gibbs point processes. This proposal has its roots in the **MPPLIB** library developed mainly by [2] where exact simulation algorithms for marked point processes were translated into **C++** in order to perform massive simulation studies. The current version of the library is freely available at the following address <https://gitlab.univ-lorraine.fr/labs/iecl/drlib>.

Interacting point process

The points are situated in W a compact region of \mathbb{R}^2 . The marks are described by the probability space (M, \mathcal{M}, ν_M) .

Unit Intensity Homogeneous Poisson Point Process with independent marks : independence - completely random patterns

- the number of points is a Poisson r.v. $\sim \mathcal{P}(\nu(W))$ with $\nu(W)$ the Lebesgue measure of W
- points are distributed uniformly in W while marks are associated to each points i.i.d. w.r.t. ν_M

Interactions can be specified via a probability density w.r.t. the previous process.

Strauss process : repulsive patterns Its probability density is

$$p(\mathbf{x}) \propto \beta^{n(\mathbf{x})} \gamma^{s_r(\mathbf{x})}$$

with $n(\mathbf{x})$ the number of points, $s_r(\mathbf{x}) = \sum_{i,j=1, i < j}^{n(\mathbf{x})} \mathbf{1}\{d(x_i, x_j) \leq r\}$ the number of pairs of points closer than the prefixed r . The model parameters are $\beta > 0$ and $\gamma \in]0, 1]$.

Area interaction process : repulsive or clustered patterns Its probability density is

$$p(\mathbf{x}) \propto \beta^{n(\mathbf{x})} \gamma^{a_r(\mathbf{x})}$$

with $n(\mathbf{x})$ the number of points, $a_r(\mathbf{x}) = \nu[\bigcup_i n(\mathbf{x})b(x_i, r)]/\pi r^2$ the volume of the union of balls centred on the points of configuration and of radius r . The model parameters are $\beta, \gamma > 0$.

A practical marked extension: discs with random radius

Simulation

Problem: the normalising constant of the previous models is not always available in analytical closed form.

Adopted solution: use MCMC methods - simulate a Markov chain whose unique equilibrium distribution is the distribution of the point process of interest.

Algorithms: spatial birth-and-death processes, Metropolis-Hastings dynamics, Gibbs samplers, perfect algorithms

Key ideas: add or remove a point from the current configuration till equilibrium is reached. The construction of the acceptance probability for the proposed transition guarantees convergence properties of the simulation algorithms [3, 4]

Posterior sampling

Problem: sampling the posterior is difficult because the normalising constant ratio does not simplify

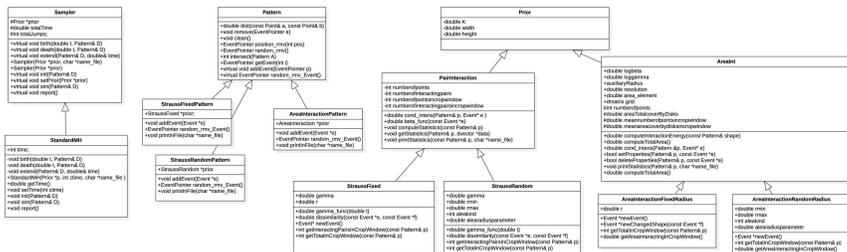
Adopted solution: ABC Shadow, SSA algorithms [5, 6].

Key ideas: approximate the behaviour of Markov chain that has the equilibrium distribution the posterior of interest.

Algorithm description: fix δ and m . Assume the observed pattern is \mathbf{y} and the current state is θ_0 .

- Generate \mathbf{x} according to $p(\mathbf{x}|\theta_0)$.
- For $k = 1$ to m do
 - Generate a new candidate ψ following the density $U_\delta(\theta \rightarrow \psi) = \frac{1}{V_\delta} \mathbb{1}_{b(\theta, \delta/2)}\{\psi\}$ with V_δ vol. of the ball $b(\theta, \delta/2)$.
 - The new state $\theta_k = \psi$ is accepted with probability $\alpha_s(\theta_{k-1} \rightarrow \theta_k) = \min\{1, \frac{f(\mathbf{y}|\theta_k)p(\theta_k)}{f(\mathbf{y}|\theta_{k-1})p(\theta_{k-1})} \frac{f(\mathbf{x}|\theta_{k-1})}{f(\mathbf{x}|\theta_k)}\}$ otherwise $\theta_k = \theta_{k-1}$.
- Return θ_m .
- If another sample is needed, go to step 1 with $\theta_0 = \theta_m$.

Implementation and classes diagram



From the basis structure, the following derived classes implement the corresponding models:

StraussFixedRadius : unmarked Strauss

StraussRandomRadius : marked Strauss - each point has a different range

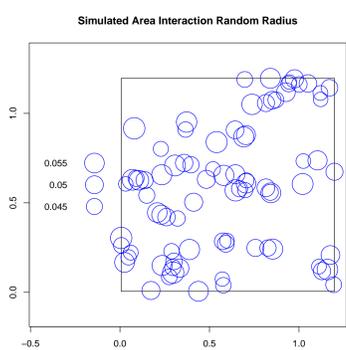
The C++ implementation builds basic classes following the mathematical principles of the problem on hand. For example:

- Event, Pattern, Prior** classes to build configurations of objects and point processes models
- Sampler, StandardMH** classes to build simulation dynamics and the Metropolis Hastings algorithm
- ABCShadow** implements the posterior sampling

AreaInteractionFixedRadius : unmarked Area Interaction

AreaInteractionRandomRadius : marked Area Interaction - each point occupies a different territory

Sample Area Int model - random radius



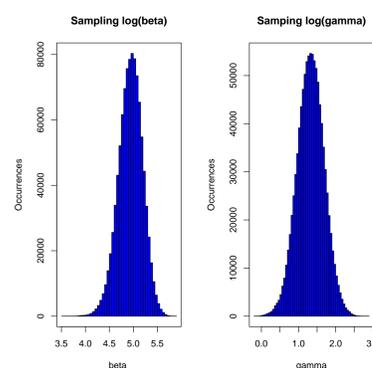
A sample realisation of the Area Interaction model - random radius

- $\log(\beta) = 5.0$; $\log(\gamma) = 1.4$
- radius \sim Uniform(0.04, 0.06)
- grid resolution(side) = 0.005

Observed sufficient statistics computed by averaging 1000 realisations:

- mean number of points = 55.61
- $a(r) = -\frac{\text{surface covered by disks}}{\pi(r_{\min} + r_{\max})/2} = -50.26$

Posterior sampling model parameters



Posterior sampling of the model parameters $(\log \beta, \log \gamma)$ with the observed sufficient statistics

$nt = 55.61$ $a(r) = -50.26$

with :

$r \in [0.04, 0.06]$

$\log(\beta) \in [3.5, 5.5]$

$\log(\gamma) \in [-5.0, 0.0]$

mh_time = 400

time_mh_theta = 100

nbiter_theta = 1000000

Next challenges

- superposition of interacting point processes multiple interactions
- inference from patterns with an increased number of points
- increase the number of dimensions
- communicate with **R spatstat** routines

Bibliography

- A. J. Baddeley, E. Rubak, R. Turner, *Spatial Point Patterns: Methodology and Applications with R* (Chapman and Hall/CRC Press, London, 2016).
- M. N. M. van Lieshout, R. S. Stoica, *Computational Statistics and Data Analysis* **51**, 679–698 (2006).
- M. N. M. van Lieshout, *Markov Point Processes and their Applications* (Imperial College Press, London, 2000).
- J. Møller, R. P. Waagepetersen, *Statistical inference and simulation for spatial point processes* (Chapman and Hall/CRC, Boca Raton, 2004).
- R. S. Stoica, A. Philippe, P. Gregori, J. Mateu, *Statistics and Computing* **27**, 1225–1238 (2017).
- R. S. Stoica, M. Deaconu, A. Philippe, L. Hurtado-Gil, *Spatial Statistics* **43**, 1–21 (2021).