

# TRANSFORMING SPATIAL POINT PROCESSES INTO POISSON PROCESSES USING RANDOM SUPERPOSITION

JESPER MØLLER\* AND

KASPER K. BERTHELSEN,\*\* Aalborg University

## Abstract

Most finite spatial point process models specified by a density are locally stable, implying that the Papangelou intensity is bounded by some integrable function  $\beta$  defined on the space for the points of the process. It is possible to superpose a locally stable spatial point process  $X$  with a complementary spatial point process  $Y$  to obtain a Poisson process  $X \cup Y$  with intensity function  $\beta$ . Underlying this is a bivariate spatial birth–death process  $(X_t, Y_t)$  which converges towards the distribution of  $(X, Y)$ . We study the joint distribution of  $X$  and  $Y$ , and their marginal and conditional distributions. In particular, we introduce a fast and easy simulation procedure for  $Y$  conditional on  $X$ . This may be used for model checking: given a model for the Papangelou intensity of the original spatial point process, this model is used to generate the complementary process, and the resulting superposition is a Poisson process with intensity function  $\beta$  if and only if the true Papangelou intensity is used. Whether the superposition is actually such a Poisson process can easily be examined using well-known results and fast simulation procedures for Poisson processes. We illustrate this approach to model checking in the case of a Strauss process.

*Keywords:* Complementary point process; coupling; local stability; model checking; Papangelou conditional intensity; spatial birth–death process; Strauss process

2010 Mathematics Subject Classification: Primary 60G55; 62M99

Secondary 62M30; 66C60; 62H11

## 1. Introduction

A spatial birth–death process is a continuous-time jump process where each jump consists of either adding or removing a point from a finite spatial point pattern. Preston [17] provided a detailed mathematical study of such processes, and showed among other things that, under suitable conditions, (approximate) realisations of a finite spatial point process can be obtained by running a spatial birth–death process for a long enough time; this point was also used in [10] and [19], and in connection to perfect simulation algorithms in [6], [11], and [12]. Spatial birth–death processes have also been used as statistical models for geological data [7], [22] and sand dunes [15], and for Bayesian analysis of mixture models with an unknown number of components [21].

Preston [17] established the existence of a spatial birth–death process through a coupling to a nonexplosive birth–death process on the nonnegative integers, which can be extended to a ‘dominating’ spatial birth–death process. This coupling is particularly useful in connection

Received 3 September 2010; revision received 27 June 2011.

\* Postal address: Department of Mathematical Sciences, Aalborg University, Fredrik Bajers Vej 7G, 9220 Aalborg Øst, Denmark.

\*\* Email address: kkb@math.aau.dk

with point processes that are locally stable, the latter being a property satisfied by most spatial point process models specified by a density; this condition and other background material are presented in Section 2. Briefly, local stability implies that the Papangelou conditional intensity  $\lambda(x, u)$  is bounded from above by an integrable function  $\beta(u)$  defined on  $S$ , where  $S$  denotes the state space of the points,  $x$  is any finite point pattern (i.e. a finite subset of  $S$ ), and  $u \in S \setminus x$  is any point. To describe the coupling construction, consider a (dominating) birth–death process  $D_t$  with birth rate  $\beta(u)$  and death rate 1 so that its equilibrium distribution is a Poisson process on  $S$  with intensity function  $\beta$ . It is possible, by a dependent thinning of  $D_t$ , to obtain a (target) birth–death process  $X_t$  with birth rate  $\lambda(x, u)$  and death rate 1 such that its distribution converges towards that of  $X$  as time  $t$  tends to  $\infty$ : the dependent thinning is such that if  $X_0 \subseteq D_0$  then  $X_t \subseteq D_t$  for all  $t > 0$  (explaining what is meant by ‘dominating’). Further details are given in Section 2.2. This coupling construction also plays a key role in connection with the perfect simulation algorithms of locally stable point processes given by the dominating coupling-from-the-past (CFTP) algorithm [11], [12] and the method of clans of ancestors [6].

In this paper we study the birth–death process  $Y_t = D_t \setminus X_t$ , i.e. the points in the dominating process  $D_t$  that are *not* included in the target birth–death process  $X_t$ . We refer to  $Y_t$  as the *complementary* birth–death process (to the target birth–death process  $X_t$ ). In Section 2.2 we define the bivariate birth–death process  $(X_t, Y_t)$ , and in Section 3 we establish that  $(X_t, Y_t)$  converges towards a bivariate point process  $(X, Y)$ , where we call  $Y$  the *complementary* point process (to the target point process  $X$ ). In general, it seems difficult to say anything detailed about this equilibrium distribution except in the special cases considered in Section 3 and in Appendix A.

Although the distribution of  $Y$  conditional on  $X = x$  seems complicated in general, it turns out to be simple to simulate from this conditional distribution. In Section 4.1 we present an algorithm which is both fast and easily implemented. In Section 4.2 we study the speed of the algorithm, which, unlike dominating CFTP, depends only on  $\beta$ , i.e. it does not depend on any (interaction or monotonicity) properties of  $\lambda$ , except on its upper bound  $\beta$ .

The algorithm may be used for model checking: given data  $x$  (a finite point pattern in  $S$ ) and a model for the Papangelou intensity of the underlying spatial point process  $X$ , this model is used for generating a realisation  $y$  from the complementary process conditional on  $X = x$ . In Section 5.1 we establish that the resulting superposition  $x \cup y$  is a Poisson process with intensity function  $\beta$  if and only if the true Papangelou intensity is used. Whether the superposition is actually such a Poisson process can easily be examined using theoretical results for (functional) summary statistics of Poisson processes, where quantiles of the summary statistics can be quickly simulated. In Section 5.2 we illustrate this approach to model checking in the case of a Strauss process [10], [23].

The above model checking procedure of superimposing the complementary point pattern on the data pattern and checking if the resulting point pattern is Poisson has some similarities to the approach considered by Møller and Schoenberg [14]. Their procedure is based on dependent thinning of the data pattern  $x$ , obtaining a realisation of a Poisson process if the assumed model for  $X$  is correct. This construction relies on an assumption of a positive lower bound on the Papangelou intensity on  $S$  which is typically not available for most point processes of interest.

## 2. Preliminaries

### 2.1. Assumptions

For simplicity and specificity, we consider a spatial point process  $X$  defined on a Borel set  $S \subset \mathbb{R}^k$  ( $k \in \{1, 2, \dots\}$ ) of finite and positive Lebesgue measure  $|S|$ , where, with probability 1,

$X$  is finite and simple (i.e. has no multiple points). This means that  $X$  can be considered as a random finite subset of  $S$ . So we let the state space  $\Omega$  of  $X$  be the set of all finite point configurations contained in  $S$ , i.e.

$$\Omega = \{x \subset S : n(x) < \infty\},$$

where  $n(x)$  is the cardinality of  $x$  (for  $n(x) = 0$ ,  $x = \emptyset$  is the empty point configuration). For measure theoretical details, see, e.g. Appendix B of [16]. The setting covers most cases of practical interest, but our methods can easily be extended to nonsimple point processes defined on a general state space and using an exponential state space setting [5], [17], [20]. We refer to  $X$  as our *target point process*.

Let  $\beta$  be a nonnegative Lebesgue integrable function defined on  $S$ , denote by  $\text{Poisson}(S, \beta)$  the distribution of the Poisson process on  $S$  with intensity function  $\beta$ , and set  $\nu = \text{Poisson}(S, 1)$  (the distribution of the homogeneous Poisson process on  $S$  with intensity 1). Note that, for any nonnegative measurable function  $h$  defined on  $\Omega$ ,

$$\int h(x) \, d\nu(x) = e^{-|S|} h(\emptyset) + \sum_{n=1}^{\infty} \frac{e^{-|S|}}{n!} \int_S \cdots \int_S h(\{x_1, \dots, x_n\}) \, dx_1 \cdots dx_n. \tag{1}$$

We assume that  $X$  is absolutely continuous with respect to  $\nu$  and denote its density by  $f$ . We also assume that  $f$  is *locally stable* with respect to  $\beta$ , i.e. for any  $x \in \Omega$  and any  $u \in S \setminus x$ ,

$$f(x \cup \{u\}) \leq \beta(u) f(x). \tag{2}$$

This condition is satisfied for most point process models specified by a density (where of course the choice of  $\beta$  depends on the density); see [8], [16], and the references therein. Clearly, (2) implies that the *Papangelou conditional intensity* defined, for any  $x \in \Omega$  and any  $u \in S \setminus x$ , by

$$\lambda(x, u) = \frac{f(x \cup \{u\})}{f(x)} \quad \left( \text{taking } \frac{0}{0} = 0 \right) \tag{3}$$

is bounded by  $\beta(u)$ . In fact, local stability also implies that many desirable properties for simulation algorithms are satisfied; cf. [16] and the references therein. Note that

$$b = \int_S \beta(u) \, du \tag{4}$$

is finite and equal to the mean number of points under  $\text{Poisson}(S, \beta)$ . Henceforth, to avoid the trivial case where  $f(x) = 0$  whenever  $x \neq \emptyset$ , we assume that  $b > 0$ .

### 2.2. Coupled spatial birth–death processes

We shall exploit the fact that (3) ensures a coupling of a continuous-time spatial birth–death process  $\{X_t : t \geq 0\}$  with a dominating spatial birth–death process  $\{D_t : t \geq 0\}$  such that  $X_t \subseteq D_t$  for all times  $t \geq 0$ , where the  $X_t$  process has birth rate  $\lambda(x, u)$  and death rate 1, and the  $D_t$  process has birth rate  $\beta(u)$  and death rate 1. Both processes are time reversible,  $X_t$  has equilibrium density  $f$ , and the equilibrium distribution of  $D_t$  is  $\text{Poisson}(S, \beta)$ . See [11], [12], [16, Appendix G], and [17]. However, in general, as shown in Section 3, the coupled process  $(X_t, D_t)$  is not time reversible.

For later use, we now recall the details of the coupling construction, where we let the initial states be arbitrary, except that it is assumed that  $X_0 \subseteq D_0$ .

First, we generate the dominating birth–death process  $\{D_t : t \geq 0\}$  as follows. We start by generating a pure birth process on  $S$  with birth rate  $\beta(u)$ . Viewed as a space–time point process, this is simply a Poisson process on  $S \times [0, \infty)$  with intensity function  $\rho(u, t) = \beta(u)$ . For each space–time Poisson point  $(u, t)$ , we generate a lifetime  $\tau(u)$  which is exponentially distributed with mean 1; these lifetimes are independent of the birth process, and the lifetimes are mutually independent. In the dominating birth–death process, the point  $u$  is then included for the time period starting at the birth time  $t$  and ending at the death time  $t + \tau(u)$  (where  $u$  is excluded).

Second, the  $X_t$  process is obtained from the  $D_t$  process by a dependent thinning. Write  $D_{t-}$  and  $X_{t-}$  for the states of the processes just before time  $t$ . If a birth happens in the dominating spatial birth–death process at time  $t > 0$  so that  $D_t = D_{t-} \cup \{u\}$ , then, conditional on this knowledge and what previously has happened in the two processes, with probability  $\lambda(X_{t-}, u)/\beta(u)$ , set  $X_t = X_{t-} \cup \{u\}$ , otherwise do nothing, i.e.  $X_t = X_{t-}$  is unchanged. Moreover, if a death happens in the dominating spatial birth–death process at time  $t > 0$  so that  $D_t = D_{t-} \setminus \{u\}$ , where  $u \in D_{t-}$ , then  $X_t = X_{t-} \setminus \{u\}$  (of course,  $X_t = X_{t-}$  is unchanged if  $u$  is not in  $X_{t-}$ ). Finally, as a transition in the  $X_t$  process can only happen if a similar transition happens in the  $D_t$  process, it follows that  $X_t \subseteq D_t$  for all  $t \geq 0$ .

We call  $Y_t = D_t \setminus X_t$ ,  $t \geq 0$ , the complementary spatial birth–death process. Note that  $\{(X_t, Y_t) : t \geq 0\}$  is a bivariate jump process such that a transition from a given state

$$(x, y) = (\{x_1, \dots, x_m\}, \{y_1, \dots, y_n\}) \in \Omega \times \Omega$$

of the process (with  $x = \emptyset$  if  $m = 0$ , and  $y = \emptyset$  if  $n = 0$ ) can be one of the following four types. The rate of transition is

- (i)  $\lambda(x, u)$  if  $(x \cup \{u\}, y)$  is the new state, i.e. when a birth of a point  $u \in S$  happens in the  $D_t$  process and it is accepted into the  $X_t$  process;
- (ii)  $\beta(u) - \lambda(x, u)$  if  $(x, y \cup \{u\})$  is the new state, i.e. when a birth of a point  $u \in S$  happens in the  $D_t$  process and it is not accepted into the  $X_t$  process;
- (iii) 1 if  $(x \setminus \{x_i\}, y)$  is the new state, i.e. when the  $i$ th point in the  $X_t$  process dies ( $i \in \{1, \dots, m\}$  and provided  $m > 0$ );
- (iv) 1 if  $(x, y \setminus \{y_j\})$  is the new state, i.e. when the  $j$ th point in the  $Y_t$  process dies ( $j \in \{1, \dots, n\}$  and provided  $n > 0$ ).

### 3. The equilibrium distribution of the bivariate jump process

The  $(X_t, Y_t)$  process converges in distribution towards a unique equilibrium distribution  $\Pi$ ; in fact, the process converges geometrically fast towards  $\Pi$  as seen by combining the results of [13] with those in Appendix G of [16]. Henceforth, assume that  $(X, Y)$  follows  $\Pi$ . We call  $Y$  the *complementary point process* (to the target point process  $X$ ). Note that  $D = X \cup Y$  follows Poisson( $S, \beta$ ), but what else can we say about  $\Pi$ ? Propositions 1–3 below are verified in Appendix A.

**Proposition 1.** *The equilibrium distribution  $\Pi$  is absolutely continuous with respect to the product measure  $\nu \times \nu$ .*

We need some further notation. Let  $\pi(x, y)$  denote the density of  $\Pi$  with respect to  $\nu \times \nu$ . Recall that, for  $x \in \Omega$ ,  $n(x)$  denotes the cardinality of  $x$ . If  $n(x) = 0$ , set  $\sum_{u \in x} q(x, u) = 0$  for any real function  $q(x, u)$ .

**Proposition 2.** *Apart from a  $(\nu \times \nu)$ -nullset, the equilibrium density  $\pi$  is the unique density satisfying the equation*

$$\begin{aligned}
 & [b + n(x) + n(y)]\pi(x, y) \\
 &= \sum_{u \in x} \lambda(x \setminus \{u\}, u)\pi(x \setminus \{u\}, y) + \sum_{u \in y} [\beta(u) - \lambda(x, u)]\pi(x, y \setminus \{u\}) \\
 &+ \int_S \pi(x \cup \{u\}, y) \, du + \int_S \pi(x, y \cup \{u\}) \, du
 \end{aligned} \tag{5}$$

for all  $(x, y) \in \Omega \times \Omega$ .

We have not been able to solve (5) without imposing rather restrictive conditions, such as in Proposition 3 below or as in the examples discussed in Appendix A.

One attempt at solving (5) is given by solving the detailed balance condition

$$\pi(x, y)\lambda(x, u) = \pi(x \cup \{u\}, y), \quad \pi(x, y)(\beta(u) - \lambda(x, u))\pi(x, y \cup \{u\}), \tag{6}$$

which is equivalent to time reversibility of the  $(X_t, Y_t)$  process. This is, however, only satisfied in the following simple case.

**Proposition 3.** *The equilibrium density  $\pi(\cdot, \cdot)$  satisfies the detailed balance condition (6) if and only if  $\lambda(x, u) = \lambda(u)$  does not depend on  $x$ , in which case  $X$  and  $Y$  are independent Poisson processes on  $S$  with intensity functions  $\lambda(u)$  and  $\beta(u) - \lambda(u)$ , respectively.*

As noted in Remark 1 (Section 5.1) and in Appendix A, apart from the case where the detailed balance condition (6) holds, the conditional distribution of  $Y$  given  $X = x$  is in general a complicated distribution—nevertheless, we can easily simulate from this conditional distribution, as shown in Section 4.1. So, in general, it seems difficult to explicitly evaluate the joint density of  $X$  and  $Y$ . Also, the density of  $Y$  seems in general to be very complicated, as discussed in Appendix A.

## 4. Conditional simulation of the complementary point process

### 4.1. Simulation procedure

The following algorithm provides an easy way to make a conditional simulation  $Y(x)$  of the complementary point process, given that  $x$  is a realization from the target point process  $X$ .

**Algorithm 1.** Inputs:  $S \subset \mathbb{R}^d$ , where  $S$  has a finite and positive volume;  $f$ , a locally stable density for a target point process on  $S$ ;  $\lambda$ , the corresponding Papangelou conditional intensity;  $\beta(u)$ , an upper bound on  $\lambda(\cdot, u)$ ;  $x \subset S$ , a finite point configuration.

- (a) Set  $Y(x) = \emptyset$  and generate  $Z$  from  $\text{Poisson}(S, \beta)$ . If  $Z = \emptyset$  then set  $T = 0$  and go to (e).
- (b) For each point  $u \in Z$ , generate an exponentially distributed lifetime  $T_u$  with mean 1, and a uniformly distributed ‘mark’  $M_u$  on  $[0, 1]$ , assuming that all these times and marks are mutually independent. Set  $T = \max\{T_u : u \in Z\}$ .
- (c) Set  $X_0 = x$  and generate the spatial birth–death process  $X_t$  with birth rate  $\lambda$  and death rate 1, stopping the generation at time  $T$ , assuming that this generation, conditional on  $T$ , is independent of everything else in (a)–(b).

- (d) For each  $u \in Z$ , if  $M_u > \lambda(X_{T_u}, u)/\beta(u)$ , add  $u$  to  $Y(x)$ , i.e.  $Y(x) \leftarrow Y(x) \cup \{u\}$ .
- (e) Return  $Y(x)$ .

**Theorem 1.** *The output  $Y(x)$  in Algorithm 1 is a realization from the conditional distribution of the complementary point process  $Y$  given that  $X = x$  is a realization from the target point process.*

*Proof.* Intuitively, this follows by

- imagining that we have extended the  $(X_t, Y_t)$  process to all times  $t \in \mathbb{R}$  such that it is in equilibrium; this is easily done, since the  $(X_t, D_t)$  process regenerates each time  $D_t = \emptyset$  (see, e.g. Appendix G of [16]);
- observing that, by (i)–(iv) in Section 2.2, conditional on  $\{X_t : t \in \mathbb{R}\}$ , the births in  $\{Y_t : t \in \mathbb{R}\}$  form a space–time Poisson process  $B$  on  $S \times \mathbb{R}$  with intensity function  $\mu(u, t) = \beta(u) - \lambda(X_t, u)$ , the corresponding lifetimes are mutually independent and independent of  $B$ , and each lifetime is exponentially distributed with mean 1;
- noting that  $B$  can be obtained by an independent thinning from a space–time Poisson process on  $S \times \mathbb{R}$  with intensity function  $\rho(u, t) = \beta(u)$ , where the retention probability for a space–time point  $(u, t)$  is  $\mu(u, t)/\rho(u, t) = 1 - \lambda(X_t, u)/\beta(u)$ .

For a formal proof, it is convenient to reverse time, and to imagine that we have generated more than is actually needed, as described in the following.

Let  $\{D_t^* : t \leq 0\}$  be an independent copy of the dominating spatial birth–death process  $\{D_t : t \leq 0\}$  considered backwards in time, where  $D_0^* = Z$ . We use this notation, since (as discussed at the beginning of Section 4.2) the  $D_t$  process may instead be used when generating the  $X_t$  process. In Algorithm 1(b),  $T$  is the largest lifetime of the points in  $Z$ ; correspondingly, let  $\tilde{T}$  be the first time before time 0 where a point in  $D_0^*$  was born when the  $D_t^*$  process is considered forwards in time, setting  $\tilde{T} = 0$  if  $D_0^* = \emptyset$ ; so  $-\tilde{T}$  is distributed as  $T$ . Moreover, suppose that we have generated the  $X_t$  process backwards in time  $t \leq 0$ , independently of the  $D_t^*$  process and anything else associated to this process, as considered below, and with  $X_0 = X$ .

By time reversibility, the generation of these processes is just like running them forwards in time. To each birth time  $t$  in  $D_t^*$  (considered forwards in time), we attach a mark given by a uniformly distributed random variable  $M_t$  on  $[0, 1]$ . All these marks are assumed to be mutually independent and independent of  $\{(X_t, D_t^*) : t \leq 0\}$ .

Moreover, suppose that, for any time  $s < 0$ , we have generated a complementary spatial birth–death process  $Y_t^s$  forwards in time  $t \in [s, 0]$  in the following way. Initially,  $Y_s^s = \emptyset$ . Furthermore, a birth in the  $Y_t^s$  process can only happen if it also happens in the  $D_t^*$  process: if  $D_t^* = D_{t-}^* \cup \{u\}$  then  $Y_t^s = Y_{t-}^s \cup \{u\}$  if  $M_t > \lambda(X_t, u)/\beta(u)$ , and  $Y_t^s = Y_{t-}^s$  otherwise. Similarly, a death in the  $Y_t^s$  process can only happen if it also happens in the  $D_t^*$  process: if a death happens so that  $D_t^* = D_{t-}^* \setminus \{u\}$  (where  $u \in D_{t-}^*$ ) then  $Y_t^s = Y_{t-}^s \setminus \{u\}$ . Hence,  $\{(X_t, Y_t^s) : s \leq t \leq 0\}$  is seen to be a jump process with transition rates as given in (i)–(iv) in Section 2.2. Consequently,  $\{(X_t, Y_t^s) : s \leq t \leq 0\}$  is distributed as  $\{(X_t, Y_t) : 0 \leq t \leq -s\}$  with  $X_0$  in equilibrium and  $Y_0 = \emptyset$ .

Note that  $Y_0^s \subseteq D_0^*$  and whether or not a birth happens in the complementary spatial birth–death process does not depend on the history of this process. Hence, to generate  $Y_0^s$ , if  $s \leq \tilde{T}$ , we need to only consider the death times of the points in  $D_0^*$  (when  $D_t^*$  is viewed backwards in time) and to use the states of the  $X_t$  process at these death times. So in our simulation procedure we need only steps (a)–(e), and  $(X, Y(X))$  is distributed as the limiting distribution

of  $(X_{-s}, Y_{-s})$  as  $-s \rightarrow \infty$ . Therefore,  $(X, Y(X))$  follows  $\Pi$ , and so  $Y(X)$  is distributed as  $Y$  conditional on  $X$ . Thereby Theorem 1 is verified.

**Remark 1.** It follows from the proof of Theorem 1 that the conditional distribution of  $Y$  given  $\{X_t : t \geq 0\}$  is a Poisson process on  $S$  with intensity function

$$\beta(u) - \sum_{i=0}^{\infty} (e^{-\tau_i} - e^{-\tau_{i+1}}) \lambda(X_{\tau_i}, u), \quad u \in S, \quad (7)$$

where  $\tau_0 = 0$  and  $\tau_1 < \tau_2 < \dots$  denote the transition (or jump) times of  $\{X_t : t \geq 0\}$ , and  $e^{-\tau_i} - e^{-\tau_{i+1}}$  is the probability that an exponentially distributed lifetime with mean 1 falls in the interval from  $\tau_i$  to  $\tau_{i+1}$  (within this interval  $X_t$  is constant). Now,  $Y$  conditional on  $X = x$  is distributed as  $Y$  conditional on  $X_0 = x$ , and the latter distribution may in principle be obtained by considering the Poisson process on  $S$  with intensity function (7) and integrating over all possible paths of  $\{X_t : t > 0\}$  when  $X_0 = x$ . However, apart from the special case where  $\lambda(x, u) = \lambda(u)$  does not depend on  $x$ , this calculation appears to be very complicated, indicating that the conditional distribution of  $Y$  given  $X$  is in general a complicated distribution.

**Remark 2.** A comparison of Algorithm 1 with perfect simulation algorithms seems in order, since the proof of Theorem 1 has some similarity to arguments used when establishing the correctness of the CFTP algorithm in [18], the dominating CFTP algorithm in [11] and [12], and the method of clans of ancestors in [6]. The latter two algorithms are used for perfect simulation of a locally stable point process, using spatial birth–death processes in different ways. As argued below, Algorithm 1 is much simpler to implement and much faster than these perfect simulation algorithms.

The speed of the dominating CFTP algorithm depends much on the monotonicity properties of  $\lambda(x, u)$  (used for generating a sequence of so-called lower and upper processes), and how strong the interaction is between  $u$  and neighbouring points in  $x$  (a point  $v \in x$  is said to be a neighbour to  $u$  if  $\lambda(x, u)$  depends on  $v$ ). In fact, in cases where  $\lambda(x, u)$  can be much smaller than  $\beta(u)$ , the dominating CFTP algorithm can be very slow [3]. Moreover, a doubling scheme is used in the dominating CFTP algorithm (this doubling scheme is for the abovementioned sequence of lower and upper processes), where Berthelsen and Møller [3] recommend that the first time in the doubling scheme should be random and distributed as  $T$  (more precisely, this is the case if the waiting times for transitions are included; in fact, we need only consider the jump chain in the dominating CFTP algorithm, so this partly reduces the computations). Furthermore, the method of clans of ancestors depends on how  $\lambda(x, u)$  specifies which points are neighbours (but not on how strong the interaction is), and this method can be very slow, in fact even slower than dominating CFTP [3], [12].

In contrast, Algorithm 1 depends neither on any monotonicity property of  $\lambda(x, u)$ , on how strong the interaction is, nor on how  $\lambda(x, u)$  specifies which points are neighbours. The speed of our simulation procedure Algorithm 1(a)–(e) depends only on  $b$  (defined in (4)), as further discussed in Section 4.2.

**Remark 3.** Algorithm 1 is useful when verifying Theorem 1. In practice, it is easier to use the following algorithm, where we exploit the fact that the actual lifetimes in Algorithm 1 are not needed if we know the order in which the points are born and die. Moreover, we can simultaneously generate the jumps in (b)–(d) of Algorithm 1. The output of the following algorithm is therefore still a conditional simulation of  $Y$  given  $X = x$ .



**Algorithm 2.** Inputs:  $S \subset \mathbb{R}^d$ , where  $S$  has a finite and positive volume;  $f$ , a locally stable density for a target point process on  $S$ ;  $\lambda$ , the corresponding Papangelou conditional intensity;  $\beta(u)$ , an upper bound on  $\lambda(\cdot, u)$ ;  $x \subset S$ , a finite point configuration.

- (a) Set  $Y(x) = \emptyset$  and  $w = x$ . Generate  $M$  from a Poisson distribution with mean  $b$ .
- (b) While  $M > 0$ ,
  - (i) set  $n = n(w)$  and  $m = M$ , and generate a uniformly distributed variable  $v$  on  $[0, 1]$ ;
  - (ii) if  $v < m/(m + n + b)$ ,
    - (ii.1) reduce  $M$  by 1, i.e.  $M \leftarrow M - 1$ ;
    - (ii.2) generate a point  $u$  on  $S$  with density  $\beta(\cdot)/b$ ;
    - (ii.3) with probability  $1 - \lambda(w, u)/\beta(u)$ , add  $u$  to  $Y(x)$ , i.e.  $Y(x) \leftarrow Y(x) \cup \{u\}$ ;
  - (iii) if  $v \in [m/(m + n + b), (m + n)/(m + n + b)]$ ,
    - (iii.1) remove a point  $u$  from  $w$  chosen uniformly at random, i.e.  $w \leftarrow w \setminus \{u\}$ ;
  - (iv) if  $v > (m + n)/(m + n + b)$ ,
    - (iv.1) generate a point  $u$  on  $S$  with density  $\beta(\cdot)/b$ ;
    - (iv.2) with probability  $\lambda(w, u)/\beta(u)$ , add  $u$  to  $w$ , i.e.  $w \leftarrow w \cup \{u\}$ .
- (c) Return  $Y(x)$ .

**4.2. The speed of the algorithm**

In this section we consider the computational load of using Algorithm 2.

Inspecting Algorithm 2 we note that essentially it involves only two computational aspects: generating a point on  $S$  with density  $\beta(\cdot)/b$  and evaluating  $\lambda(w, u)/\beta(u)$ . We claim that, for most applications, the computational load of the latter is (much) higher than the former. In particular, this is so if  $\beta(\cdot)$  is constant, the calculation of  $\lambda(w, u)$  involves a loop over all points in  $w$  (as, e.g. in the Strauss process considered in Section 5.2), and  $n(w)$  is not (very) small. We quantify therefore the computational load of Algorithm 2 in terms of the number of times  $\lambda(w, u)/\beta(u)$  is evaluated. Algorithm 2 evaluates  $\lambda(w, u)/\beta(u)$  only in step (ii.3), which happens  $M$  times, and in step (iv.2). Letting  $N$  denote the number of times (iv.2) is evaluated, the computational load is  $C = M + N$ . Step (iv) corresponds to a birth in the dominating process  $D_t$  used in the (backwards) construction of  $X_t$  in step (c) of Algorithm 1. Hence,  $N$  corresponds to the number of births in the dominating process in a time interval of random length  $T$ , where  $T$  is defined as in Algorithm 1. For this reason, although  $T$  is not used in Algorithm 2, the mean of  $T$  is related to the mean of  $C$ , as stated in the following proposition.

**Proposition 4.** *The mean computational load is*

$$E(C) = b(1 - e^{-b}) + b E(T), \tag{8}$$

where

$$E(T) = (1 - e^{-b}) \left[ (1 - e^{-b}) \ln(b) - \int_0^b \ln(s) e^{-s} ds \right] \tag{9}$$

$$\leq (1 - e^{-b}) [(1 - e^{-b}) \ln(b) + a] \tag{10}$$

and  $a = - \int_0^1 \ln(s) e^{-s} ds \approx 0.7966$ .



*Proof.* Conditional on  $T$ , if  $T = 0$  then  $M = N = 0$ , and if  $T > 0$  then  $M$  and  $N$  are independent and Poisson distributed, with mean  $b$  and  $bT$ , respectively. Hence, as  $\exp(-b)$  is the probability that  $T = 0$ ,

$$E(C) = E(E(1[T > 0]C \mid T)),$$

where  $1[\cdot]$  is the indicator function, and so we obtain (8). Furthermore, when  $T > 0$ ,  $T$  is the maximum of  $M$  independent exponential random variables with mean 1. So, for  $t > 0$ , conditioning on  $M$ ,

$$P(T \leq t \mid T > 0) = E((1 - e^{-t})^M) = \exp(-be^{-t}),$$

so that

$$\begin{aligned} E(T) &= P(T > 0) E(T \mid T > 0) \\ &= (1 - e^{-b}) \int_0^\infty P(T > t \mid T > 0) dt \\ &= (1 - e^{-b}) \int_0^\infty (1 - \exp(-be^{-t})) dt \\ &= (1 - e^{-b}) \int_0^b \frac{1 - e^{-s}}{s} ds, \end{aligned}$$

using the substitution  $s = be^{-t}$ . Then integrating by parts gives (9), and (10) is easily obtained.

The integral in (9) can easily be evaluated by numerical integration, and in most applications, the term  $\exp(-b)$  appearing in (8), (9), and (10) will be effectively 0. It follows that

$$E(C) \leq b + b(\ln(b) + a),$$

where in most applications, ' $\leq$ ' can be replaced by ' $\approx$ '.

### 5. Model checking

#### 5.1. The random superposition procedure

Suppose that a realization  $x$  from a spatial point process  $X^*$  with 'true' density  $f^*$  is observed, and we want to check the goodness of fit for a fitted model with density  $f$ . In order to apply Algorithm 2, assume that both  $f^*$  and  $f$  are locally stable with respect to (the same)  $\beta$ . Suppose furthermore that we have generated a realisation of  $Y(X^*)$  using Algorithm 2 which has  $f$  (and not  $f^*$  unless  $f = f^*$ ) as input. By Theorem 1, the *random superposition procedure*

$$D^* = X^* \cup Y(X^*)$$

is a realization of  $\text{Poisson}(S, \beta)$  if  $f = f^*$ . Conversely, the following theorem establishes that if  $f$  and  $f^*$  do not specify the same model, then  $D^*$  does not follow  $\text{Poisson}(S, \beta)$ .

**Theorem 2.** *Assume that  $f^*$  and  $f$  are locally stable with respect to  $\beta$ . Then  $D^*$  follows  $\text{Poisson}(S, \beta)$  if and only if  $f$  and  $f^*$  agree, except on a  $v$ -nullset.*

*Proof.* We have already noted that the 'if' part holds; to verify the 'only if' part, we first observe the following. For any nonnegative measurable function  $h(x, y)$ , defined for  $(x, y) \in \Omega \times \Omega$ ,

$$\int \sum_{x \subseteq z} h(x, z \setminus x) dv(z) = e^{|S|} \int \int h(x, y) dv(x) dv(y). \tag{11}$$

To verify this, let  $x = \{x_1, \dots, x_m\}$ ,  $y = \{y_1, \dots, y_n\}$ , and  $z = x \cup y = \{z_1, \dots, z_k\}$ , and use (1) to conclude that the right-hand side of (11) equals

$$\begin{aligned} & \sum_{m=0}^{\infty} \sum_{n=0}^{\infty} \frac{e^{-|S|}}{m! n!} \int_S \cdots \int_S \int_S \cdots \int_S h(x, y) dx_1 \cdots dx_m dy_1 \cdots dy_n \\ &= \sum_{k=0}^{\infty} \sum_{m=0}^k \frac{e^{-|S|}}{k!} \int_S \cdots \int_S \sum_{\{x \subseteq z : n(x)=m\}} h(x, z \setminus x) dz_1 \cdots dz_k, \end{aligned}$$

which, by (1), reduces to the left-hand side of (11). Now, if  $X_1$  and  $X_2$  are spatial point processes on  $S$  such that  $X_1$  has density  $\pi_1$  and  $X_2$  conditional on  $X_1 = x$  has density  $\pi_2(\cdot | x)$ , then, using (11) with  $h(x, y) = \pi_1(x)\pi_2(y | x)$ , we see that  $X_1 \cup X_2$  has density

$$\pi(z) = e^{-|S|} \sum_{x \subseteq z} \pi_1(x)\pi_2(z \setminus x | x). \tag{12}$$

Let

$$q(z) = e^{|S|-b} \prod_{u \in z} \beta(u)$$

denote the density of  $Z \sim \text{Poisson}(S, \beta)$ . By (a)–(c) of Algorithm 1,  $Y(x)$  has a density  $g(\cdot | x)$ ; specifically,

$$g(y | x) = \int q(y \cup w)h(x, y, w) dv(w),$$

where

$$h(x, y, w) = E\left(\left\{\prod_{u \in y} \left(1 - \frac{\lambda(X_{T_u}, u)}{\beta(u)}\right)\right\} \left\{\prod_{u \in w} \frac{\lambda(X_{T_u}, u)}{\beta(u)}\right\}\right),$$

with the expectation calculated conditional on  $X_0 = x$ . In particular,

$$g(\emptyset | x) \geq P(Z = \emptyset) = e^{-b} > 0.$$

Moreover, by Theorem 1 and (12), for  $\nu$ -almost all  $z \in \Omega$ ,

$$q(z) = e^{-|S|} \sum_{x \subseteq z} f(x)g(z \setminus x | x). \tag{13}$$

This means that, for  $\nu$ -almost all  $z \in \Omega$ ,  $f(z)$  is determined by  $q$  and  $g$ , since first (13) with  $z = \emptyset$  gives

$$f(\emptyset) = \frac{e^{|S|}q(\emptyset)}{g(\emptyset | \emptyset)}, \tag{14}$$

and second, by induction,  $f(z)$  is given in terms of those  $f(x)$  with  $x$  strictly contained in  $z$ , using the fact that, by (13),

$$f(z) = e^{|S|} \left( q(z) - e^{-|S|} \sum_{x \subset z} f(x)g(z \setminus x | x) \right) / g(\emptyset | z). \tag{15}$$

Similarly, if  $D^*$  follows  $\text{Poisson}(S, \beta)$  then, for  $\nu$ -almost all  $z \in \Omega$ , we also obtain (13), but with  $f$  replaced by  $f^*$ , and, hence, (14)–(15), but with  $f$  replaced by  $f^*$ . Hence, the ‘only if’ part is verified.

**Remark 4.** In [14] model checking is based on a random thinning method—where it is assumed that  $\lambda(x, u) \geq \beta(u)$ , a condition which is rarely satisfied for point process models, as discussed in [14], and where the method of clans of ancestors (discussed in Remark 2) plays a key role. This seems a less appealing procedure than our random superposition procedure, since the latter is faster, simpler, and generally applicable.

## 5.2. Example: the Strauss process

In this section we consider an example of how to utilise Theorem 2 for model checking. The basic idea is as follows. Given data  $x$  generated from the ‘true’  $f^*$ , we generate a realisation  $y$  of  $Y(x)$  using Algorithm 2, which has  $f$  as the ‘input density’. Suppose that  $f$  and  $f^*$  are locally stable. According to Theorem 2 the union  $x \cup y$  is (approximately) a realisation of a Poisson process on  $S$  with intensity function  $\beta$  if and only if  $f$  and  $f^*$  specify the same model; here we write ‘approximately’ because in practice  $f$  has been estimated from the data  $x$ . The model check consists in testing the hypothesis that  $x \cup y$  is a realisation of a Poisson process on  $S$  with intensity function  $\beta$ .

There exist numerous ways of testing if a given point pattern is a realisation of a Poisson point process. In the sequel we assume that  $\beta$  is constant and restrict attention to methods based on Besag’s  $L$  function, which is a useful transformation of Ripley’s  $K$  function [4]. Informally,  $L(r)$  is a nonnegative functional point process summary which indicates the extent to which a given stationary point process exhibits clustering or repulsion at an interpoint distance  $r > 0$ . In the present context the most important property of the  $L$  function is that, for a stationary Poisson process,  $L(r) = r$ . Furthermore, for a point process with  $L(r) < r$ , the expected number of points within a distance  $r$  from a ‘typical point’ is lower than what is expected under a stationary Poisson process, indicating repulsion between the points if  $r$  is small. Similarly,  $L(r) > r$  implies that more points are expected within distance  $r$  from a typical point when compared to a stationary Poisson process, indicating clustering between the points if  $r$  is small. For more details, including extensions to the case where  $\beta$  is not constant, see [16] and the references therein.

We let  $\hat{L}(r) = \hat{L}(r; z)$  denote an estimate of  $L$  based on a point pattern  $z$  obtained by observing a stationary point process within  $S$ ; see, e.g. [9] and [16]. Usually, this estimate involves first estimating the intensity of the process; below, unless otherwise stated, we make use of the fact that the intensity is given by  $\beta$ , which is assumed to be known. If  $z$  is a realisation of a stationary Poisson process restricted to  $S$ , we expect that  $\hat{L}(r; z) - r \approx 0$  for all  $r > 0$ . Accordingly, if  $\hat{L}(r; x \cup y) - r$  deviates too much from 0, we have an indication that the model specified by  $f$  is not (close to) the true model. To get a handle on the ‘too much’ part, we need to take into account the variation in  $\hat{L}$ . Let  $\underline{L}(r) - r$  and  $\overline{L}(r) - r$  denote estimated 2.5% and 97.5% quantiles of  $\hat{L}(r; W) - r$  when  $W$  follows  $\text{Poisson}(S, \beta)$ . These estimates are based on independent simulations from  $\text{Poisson}(S, \beta)$ ; unless otherwise stated, we use  $k = 239$  such simulations  $W^{(1)}, \dots, W^{(k)}$  so that  $\underline{L}(r)$  is the fifth smallest and  $\overline{L}(r)$  the fifth largest among  $\hat{L}(r; W^{(1)}), \dots, \hat{L}(r; W^{(k)})$ . We refer to the pair of functions  $\underline{L}(r) - r$  and  $\overline{L}(r) - r$  as the (pointwise) estimated 95% Poisson envelopes. If  $\hat{L}(r; x \cup y) - r$  deviates too much outside these envelopes, we reject the assumed model. For more details on this test procedure, see, e.g. Section 7.4.1 of [9].

We now consider model checking procedures based on the random superposition procedure. For specificity, we consider a planar Strauss process with density

$$f(x) \propto \beta^{n(x)} \gamma^{s_R(x)},$$

where  $\beta > 0$ ,  $\gamma \in [0, 1]$ , and  $R > 0$  are parameters, and where  $s_R(x)$  is the number of point

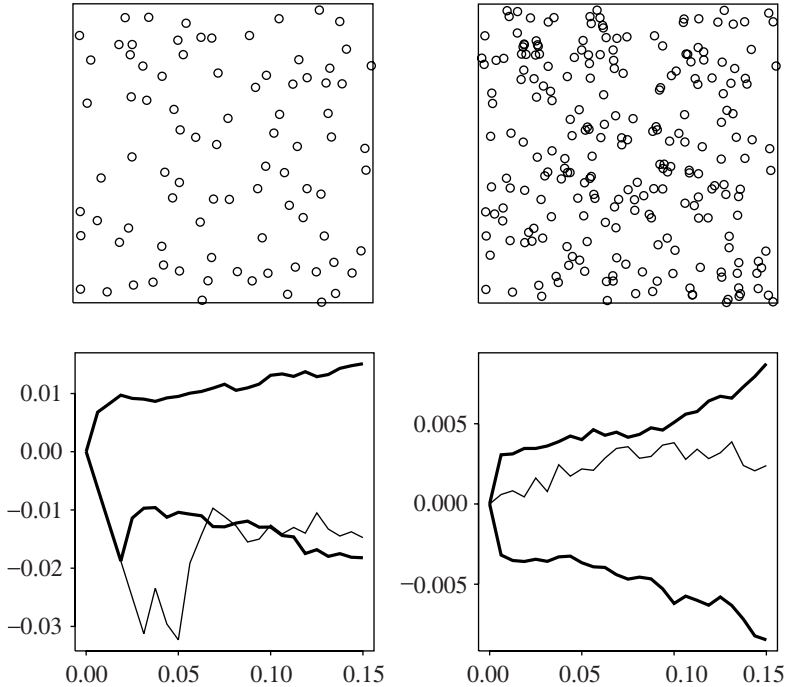


FIGURE 1: *Top left:* realisation  $x$  of a Strauss process on the unit square and with  $(\beta, \gamma, R) = (250, 0.1, 0.05)$ . *Top right:* union of  $x$  and a realisation  $y$  of the complementary process  $Y(x)$ . *Bottom left:*  $L(r; x) - r$  corresponding to top-left diagram (*thin line*). *Bottom right:*  $L(r; x \cup y) - r$  corresponding to top-right diagram (*thin line*). The thick lines in the two lower diagrams are estimated 95% Poisson envelopes.

pairs  $\{u, v\} \subseteq x$  (with  $u \neq v$ ) separated by a distance less than  $R$  [10], [23]. The Strauss process is often used in the spatial point process literature for illustrative purposes and we know how to make (even perfect) simulations under this model. It is locally stable, since

$$\lambda(x, u) = \beta \gamma^{s_R(x, u)} \leq \beta,$$

where  $s_R(x, u)$  is the number of points in  $x$  within a distance  $R$  from  $u$ . Thus,  $\beta$  is the intensity of the dominating Poisson process, while  $\gamma$  is an interaction parameter and  $R$  determines the range of interaction in the Strauss process.

The top-left diagram of Figure 1 shows a realisation  $x$  of a Strauss process on the unit square  $S = [0, 1]^2$  with  $\beta = 250$ ,  $\gamma = 0.1$ , and  $R = 0.05$ . In the following we will refer to this model as the true model. Here  $n(x) = 87$  and  $x$  is a perfect simulation obtained by the dominating CFTP algorithm described in [3]. The bottom-left diagram shows a plot of  $\hat{L}(r; x) - r$  compared to estimated 95% Poisson envelopes, where  $\underline{L}$  and  $\bar{L}$  are estimated, assuming that  $\hat{\beta} = n(x)/|S|$  (the maximum likelihood estimate under the stationary Poisson process). As  $\hat{L}(r; x) - r$  is well outside the 95% envelopes, this correctly indicates that  $x$  is not a realisation of a Poisson process. In fact, the dip in  $\hat{L}(r; x) - r$  around  $r = R$  indicates a repulsive point process. We used the `spatstat` packages for the calculation of  $\hat{L}(x)$ ; see [1] and [2].

To illustrate the use of our random superposition procedure, let  $y$  be a realisation of  $Y(x)$  under the true model. The top-right diagram of Figure 1 shows  $x \cup y$ , where  $n(x \cup y) = 235$ .

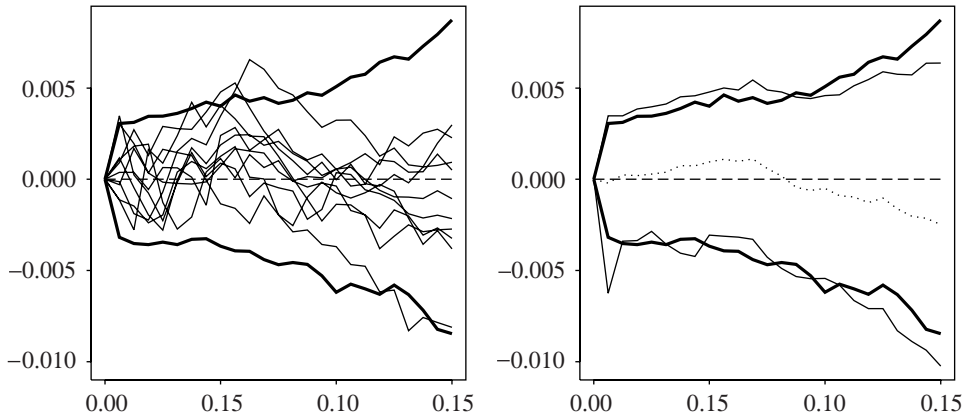


FIGURE 2: *Left:*  $\hat{L}(r; x \cup Y(x)) - r$  for ten independent realisations of  $Y(x)$ , assuming the true model. *Right:* estimated 95% envelopes for  $\hat{L}(r; x \cup Y(x)) - r$  (thin lines), together with an estimate of  $E(\hat{L}(r; x \cup Y(x)) - r)$  (dotted line). The thick lines in the two diagrams are estimated 95% Poisson envelopes for  $\hat{L}(r; W) - r$  when  $W \sim \text{Poisson}(S, \beta)$  (they are identical to the thick line in the bottom-right diagram of Figure 1).

The bottom-right diagram of Figure 1 shows  $\hat{L}(r; x \cup y) - r$ , together with estimated 95% Poisson envelopes. As  $\hat{L}(r; x \cup y) - r$  is well within these envelopes, we cannot dismiss that  $x \cup y$  is a realisation of a Poisson process. In turn this implies, correctly, that we cannot reject that the assumed model is the true model.

One may be concerned by the fact that the conclusion above of course depends on the realisation  $Y(x) = y$ . The left diagram of Figure 2 shows  $\hat{L}(r; x \cup y) - r$  for ten realisations of  $y$  from  $Y(x)$  together with estimated 95% Poisson envelopes. The right diagram of Figure 2 shows estimated 95% envelopes for  $\hat{L}(r; x \cup Y(x)) - r$ , an estimate of  $E(\hat{L}(r; x \cup Y(x)) - r)$ , and 95% Poisson envelopes. Note that the two sets of envelopes are a close match. This is in general not to be expected even if the estimated model equals the true model since, for a given  $x$ ,  $x \cup Y(x)$  is in general not distributed as  $\text{Poisson}(S, \beta)$ .

Now let us consider two misspecifications of the model. In model A we assume that  $x$  is a realisation of a Strauss process with  $\beta = 150$ ,  $\gamma = 0.5$ , and  $R = 0.05$  (i.e. incorrect  $\beta$  and  $\gamma$ , but correct  $R$ ), and in model B we assume that  $x$  is a realisation of a Strauss process with  $\beta = 125$ ,  $\gamma = 0.1$ , and  $R = 0.025$  (i.e. incorrect  $\beta$  and  $R$ , but correct  $\gamma$ ). Under both the true model and the two misspecified models, the expected number of points in the point processes are roughly the same (confirmed by simulations).

The results obtained under model A are summarised in Figure 3. Compared to Figure 2, the realisations of  $\hat{L}(r; x \cup Y(x)) - r$  are no longer centred around 0 and the two pairs of envelopes in the right diagram do not match. Since most of the  $\hat{L}(r; x \cup Y(x)) - r$  curves in the left diagram of Figure 3 are within the 95% Poisson envelopes, and keeping in mind that  $x \cup Y(x)$  is not distributed as  $\text{Poisson}(S, \beta)$ , Figure 3 is not a strong indication that the model specified in model A is wrong.

For model B, the conclusions based on Figure 4 are much clearer. The fact that both the estimated mean and estimated 2.5% envelope for  $L(r; x \cup Y(x)) - r$  are well outside the Poisson envelopes for  $r \approx 0.05$  gives clear indications of a misspecified model. Furthermore, the distinct V-shaped deviation from 0 in the curves in the left diagram of Figure 4 is unlikely under a Poisson process.

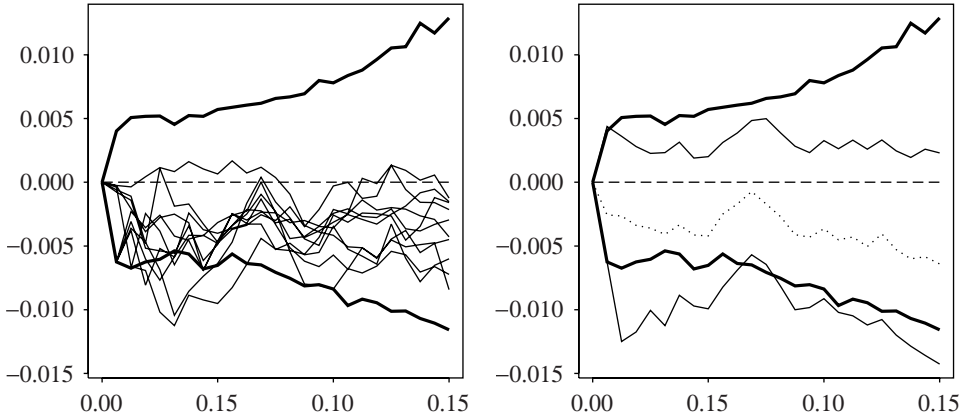


FIGURE 3: As in Figure 2, but  $\beta = 150$ ,  $\gamma = 0.5$ , and  $R = 0.05$  when generating  $Y(x)$  and  $W \sim \text{Poisson}(S, \beta)$ .

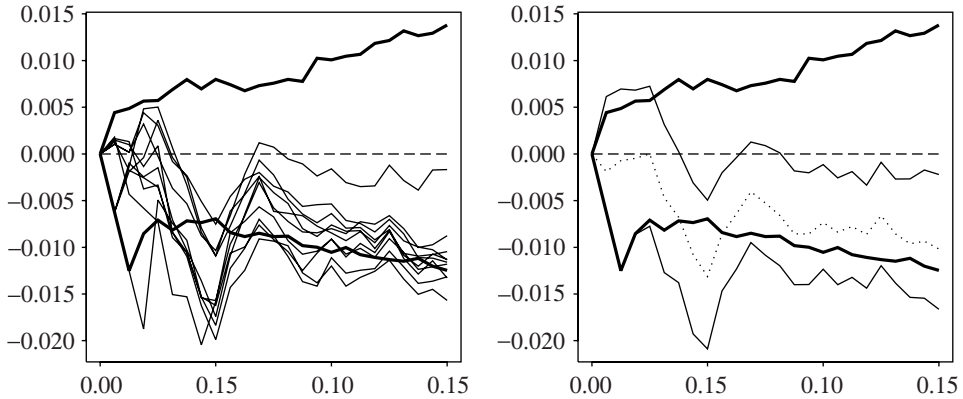


FIGURE 4: As in Figure 2, but  $\beta = 125$ ,  $\gamma = 0.1$ , and  $R = 0.025$  when generating  $Y(x)$  and  $W \sim \text{Poisson}(S, \beta)$ .

Although we prefer model checking based on functional summary statistics as illustrated in Figures 2–4, we now consider two one-dimensional test statistics given by

$$T_1(z) = \int_0^{\tilde{r}_1} (\hat{L}(r; z) - r)^2 dr$$

and

$$T_2(z) = \max_{r \in (0, \tilde{r}_2]} d(r; z) - \min_{r \in (0, \tilde{r}_2]} d(r; z), \tag{16}$$

where  $d(r; z) = (\hat{L}(r; z) - r) / (\bar{L}(r) - \underline{L}(r))$  and  $\tilde{r}_i$  is a user-specified parameter,  $i = 1, 2$ . Note that  $T_1$  captures the overall deviation from 0 expected if the model is correct, and  $T_2$  should capture large V-shaped deviations, as seen in Figure 4, while taking into account that the variance of  $\hat{L}(r; W) - r$  varies with  $r$ , where  $W \sim \text{Poisson}(S, \beta)$ . Based on 1000 realisations from  $\text{Poisson}(S, \beta)$  we obtain, for each test statistic  $T_i$ , an estimate  $\hat{T}_{i,c}$  of the critical value at the 5% significance level. For each of the misspecified models A and B, we have generated 1000 independent realisations  $x^{(1)}, \dots, x^{(1000)}$  of  $X$ , and, for each realisation  $x^{(j)}$ , we have

generated a realisation  $y^{(j)}$  from  $Y(x^{(j)})$ . For each test statistic  $T_i$ , the estimated power is given by the fraction of the 1000 realisations where the value of  $T_i(x^{(j)} \cup y^{(j)})$  is larger than  $\hat{T}_{i,c}$ .

For both models A and B and for both  $T_1$  and  $T_2$ , in order to cover well the  $V$ -shaped part of the curves in Figure 4, we have used  $\tilde{r}_1 = \tilde{r}_2 = 0.15$ . For model A, the estimated power is very low: 3.7% and 7.9% respectively for  $T_1$  and  $T_2$ . This is in agreement with the conclusion above based on a visual inspection of Figure 3. For model B, the estimated power is 11.4% and 47.7% respectively for the two test statistics. Here  $T_2$  has a reasonable level of power, which is in accordance with the conclusion based on a visual inspection of Figure 4. We expect that it is in general difficult to find a good test statistic which captures the type of deviations in Figure 4 which we noted above.

**Remark 5.** One inherent limitation of our proposed model checking procedure stems from the fact that we compare  $x \cup y$  to a homogeneous Poisson process with intensity  $\beta$ , where  $b = \beta|S|$  can be far from the maximum likelihood estimate  $n(x)$  under this Poisson process. Consider the case where  $b \gg n(x)$ , which is the case for point processes with strong interaction and dense packing, e.g. a Strauss process with  $\gamma \approx 0$  and  $R > 0$  combined with a high value of  $\beta$ . Assuming  $b \gg n(x)$  implies that  $E(n(Y(x))) \approx b \gg n(x)$ , i.e. in the union  $x \cup y$  we expect the points of the complementary point process to vastly outnumber the data points—essentially the data ‘drowns’ in the complementary point process. Consequently, the distribution of  $Y(x)$  is very similar to  $\text{Poisson}(S, \beta)$  no matter what the true model is. Hence, no matter what the true model is, the probability of rejecting a wrong model effectively equals the significance level.

**Remark 6.** A byproduct of Algorithm 2 is that when generating independent and identically distributed (i.i.d.) realisations of  $Y(x)$ , we obtain i.i.d. realisations of  $X_T$  conditional on  $X_0 = x$ . If the model is correct, any summary of  $x$  is not expected to be extreme compared to the same summary for realisations of  $X_T$ . A simple summary would be the number of points.

**Remark 7.** The conventional way to use the  $L$  function for model checking is to compare  $\hat{L}(r; x) - r$  to estimated 95% envelopes for  $\hat{L}(r; X) - r$  when  $X$  is distributed according to the assumed model (see, e.g. [9] and [16]). Obtaining the envelopes typically involves either generating i.i.d. realisations of  $X$  or subsampling from a long Markov chain converging towards the assumed model. In general, both methods are computationally more expensive (see, e.g. [16]) than generating both  $Y(x)$  and  $\text{Poisson}(S, \beta)$  used in our test. Hence, our approach has a computational advantage compared to the more conventional approach for model checking, but the cost of this computational advantage is a loss of power.

**Remark 8.** To assess the difference in power and computational load between our method and the conventional approach sketched in Remark 7, we performed a simulation experiment as follows. We now let  $X$  be distributed according to some assumed (wrong) model, e.g. model A or model B, and let the envelopes  $\underline{L}(r)$  and  $\bar{L}(r)$  be the estimated 2.5% and 97.5% quantiles for  $\hat{L}(r; X)$ . These envelopes are estimated as in the case of Poisson envelopes but with  $W^{(1)}, \dots, W^{(k)}$  replaced by  $k$  independent copies of  $X$ , denoted by  $X^{(1)}, \dots, X^{(k)}$ ,  $k = 239$ . Furthermore, let  $\tilde{L}(r) = \sum_{i=1}^k \hat{L}(r; X^{(i)})/k$  be the empirical mean estimate of  $E(\hat{L}(r; X))$ . Unlike in the Poisson case, we do not expect  $E(\hat{L}(r; X))$  to be near 0 for all values of  $r$ . With this in mind, redefine  $T_1 = \int_0^{\tilde{r}_1} (\hat{L}(r; z) - \tilde{L}(r))^2 dr$  and let  $T_2$  be as in (16), but with  $d(r; z) = (\hat{L}(r; z) - \tilde{L}(r))/(\bar{L}(r) - \underline{L}(r))$ . Each estimated critical value  $\hat{T}_{i,c}$  ( $i = 1, 2$ ) at the 5% significance level is based on 1000 independent realisations under the assumed model. Furthermore, let  $X^{*(1)}, \dots, X^{*(1000)}$  be independent realisations of  $X^*$ . The power is then estimated by the fraction of those  $T_i(X^{*(j)})$  which exceed  $\hat{T}_{i,c}$ . When  $X$  is specified by model A,



the estimated power is 60.0% for  $T_1$  and 70.9% for  $T_2$ . When  $X$  is specified by model B, the estimated power is 87.4% for  $T_1$  and 99.8% for  $T_2$ . Hence, as expected, the conventional method performs better in terms of power.

We now turn to a comparison of the computational load associated with model checking based on simulating  $Y(x)$  and the conventional model checking based on simulation of  $X$  under the fitted model. Our procedure involves generating a single realisation of  $Y(x)$  and  $k$  independent realisations of a Poisson process used for the estimation of the upper and lower Poisson envelopes. A further  $k'$ , say, independent realisations of a Poisson process are needed for estimating critical values. The conventional procedure for model checking involves generating  $k$  realisations of  $X$  using time consuming birth–death Metropolis–Hastings simulations (in fact, we are even using perfect simulations), and a further  $k'$  realisations of  $X$  when estimating a critical value. As in Section 4.2, we let the computational load be given by the number of times the Papangelou conditional intensity  $\lambda$  is evaluated; in the case of birth–death Metropolis–Hastings and perfect simulations, the computational load is also effectively given by the number of times  $\lambda$  is evaluated; see, e.g. [16]. Under the true model, using (8), the mean computational load of generating a realisation of  $Y(x)$  is 1830.4. We do not have theoretical results for the mean computational load associated with generating a realisation from the true model using perfect simulation; the average computational load from 1000 realisation of the true model is 32 448.8. Although the computational load as defined here does not give the complete picture of the practical difference in computing time, it does underline that our method for model checking has an computational advantage. The difference in mean computational load between generating  $Y(x)$  and  $X$  is less significant when using models A and B. For model A, the mean computational loads for generating  $Y(x)$  and  $X$  are 1021.6 and 2980.4, respectively. For model B, the mean loads are 828.5 and 1629.7, respectively.

In conclusion our proposed procedure is computationally advantageous compared to a more conventional way of model checking, but it comes at the price of lower power. Moreover, we find our proposed procedure informative when using envelopes for  $\hat{L}(r; x \cup Y(x)) - r$  as exemplified in the right diagrams of Figures 2–4.

### Appendix A

For  $n = 0, 1, \dots$ , define  $\Omega_n = \{x \subset S : n(x) = n\}$ . So  $\Omega = \bigcup_{n=0}^{\infty} \Omega_n$ . For any event  $F_n \subseteq \Omega_n$ , if  $n \geq 1$ ,

$$v(F_n) = \frac{e^{-|S|}}{n!} \int_S \dots \int_S 1[\{x_1, \dots, x_n\} \in F_n] dx_1 \dots dx_n. \tag{17}$$

Moreover,  $v(F_0) = 1[\emptyset \in F_0]$  if  $F_0 \subseteq \Omega_0$ , i.e. when either  $F_0$  is empty or it is the set consisting of the empty point configuration  $\emptyset$ .

*Proof of Proposition 1.* By the assumption that  $X$  is absolutely continuous with respect to  $\nu$ , it suffices to verify that, for any point configuration  $x \in \Omega_m$ , event  $F_n \subseteq \Omega_n$ , and nonnegative integers  $m$  and  $n$ ,  $P(Y \in F_n \mid X = x) = 0$  if  $\nu(F_n) = 0$ . This follows immediately from (17) and Theorem 1.

*Proof of Proposition 2.* For any events  $F_m \subseteq \Omega_m$  and  $F_n \subseteq \Omega_n$ , with  $m, n = 0, 1, \dots$ , the total rate of moving away from any state in  $F_m \times F_n$  is  $b + m + n$ ; the mean of the total rate

of moving into  $F_m \times F_n$  by a birth in the  $X_t$  process is

$$\begin{aligned}
 G_1(F_m \times F_n) &= \int_S \cdots \int_S \sum_{i=1}^m 1[\{x_1, \dots, x_m\} \in F_m, \{y_1, \dots, y_n\} \in F_n] \\
 &\quad \times \lambda(\{x_1, \dots, x_{i-1}, x_{i+1}, \dots, x_m\}, x_i) \\
 &\quad \times \pi(\{x_1, \dots, x_{i-1}, x_{i+1}, \dots, x_m\}, \{y_1, \dots, y_n\}) \\
 &\quad \times \frac{e^{-2|S|}}{m!n!} dx_1 \cdots dx_m dy_1 \cdots dy_n
 \end{aligned}$$

(setting  $G_1(F_m \times F_n) = 0$  if  $m = 0$ ); the mean of the total rate of moving into  $F_m \times F_n$  by a birth in the  $Y_t$  process is

$$\begin{aligned}
 G_2(F_m \times F_n) &= \int_S \cdots \int_S \sum_{i=1}^n 1[\{x_1, \dots, x_m\} \in F_m, \{y_1, \dots, y_n\} \in F_n] \\
 &\quad \times [\beta(y_i) - \lambda(\{x_1, \dots, x_m\}, x_i)] \\
 &\quad \times \pi(\{x_1, \dots, x_m\}, \{y_1, \dots, y_{i-1}, y_{i+1}, \dots, y_n\}) \\
 &\quad \times \frac{e^{-2|S|}}{m!n!} dx_1 \cdots dx_m dy_1 \cdots dy_n
 \end{aligned}$$

(setting  $G_2(F_m \times F_n) = 0$  if  $n = 0$ ); the mean of the total rate of moving into  $F_m \times F_n$  by a death in the  $X_t$  process is

$$\begin{aligned}
 G_3(F_m \times F_n) &= \int_S \cdots \int_S \sum_{i=1}^{m+1} 1[\{x_1, \dots, x_{i-1}, x_{i+1}, \dots, x_{m+1}\} \in F_m, \{y_1, \dots, y_n\} \in F_n] \\
 &\quad \times \pi(\{x_1, \dots, x_{m+1}\}, \{y_1, \dots, y_n\}) \\
 &\quad \times \frac{e^{-2|S|}}{(m+1)!n!} dx_1 \cdots dx_{m+1} dy_1 \cdots dy_n;
 \end{aligned}$$

and the mean of the total rate of moving into  $F_m \times F_n$  by a death in the  $Y_t$  process is

$$\begin{aligned}
 G_4(F_m \times F_n) &= \int_S \cdots \int_S \sum_{i=1}^{n+1} 1[\{x_1, \dots, x_m\} \in F_m, \{y_1, \dots, y_{i-1}, y_{i+1}, \dots, y_{n+1}\} \in F_n] \\
 &\quad \times \pi(\{x_1, \dots, x_m\}, \{y_1, \dots, y_{n+1}\}) \\
 &\quad \times \frac{e^{-2|S|}}{m!(n+1)!} dx_1 \cdots dx_m dy_1 \cdots dy_{n+1}.
 \end{aligned}$$

Consequently, by Proposition 8.1 of [17], the equilibrium distribution  $\Pi$  is the unique distribution satisfying

$$\begin{aligned}
 (b + m + n)\Pi(F_m \times F_n) &= G_1(F_m \times F_n) + G_2(F_m \times F_n) + G_3(F_m \times F_n) \\
 &\quad + G_4(F_m \times F_n)
 \end{aligned}$$

for all events  $F_m \subseteq \Omega_m$  and  $F_n \subseteq \Omega_n$ , with  $m, n = 0, 1, \dots$ . This is seen to be equivalent to

the statement in Proposition 2, since

$$\begin{aligned} \Pi(F_m \times F_n) &= \int_S \cdots \int_S 1[\{x_1, \dots, x_m\} \in F_m, \{y_1, \dots, y_n\} \in F_n] \\ &\quad \times \pi(\{x_1, \dots, x_m\}, \{y_1, \dots, y_n\}) \frac{e^{-2|S|}}{m! n!} dx_1 \cdots dx_m dy_1 \cdots dy_n. \end{aligned}$$

*Proof of Proposition 3.* The ‘if’ part is easily verified, since

$$f(x) = \exp\left(-\int_S \lambda(u) du\right) \prod_{u \in x} \lambda(u)$$

is the density of  $X$ , and

$$g(y) = \exp\left(b - \int_S \lambda(u) du\right) \prod_{u \in y} (\beta(u) - \lambda(u))$$

is the density of  $Y$ . Now suppose that (6) holds. The first equation in (6) implies that  $f(x)\lambda(x, u) = f(x \cup \{u\})$ , which is clearly satisfied and just means that  $X$  has density  $f$ . Thus,  $g(y | x) = \pi(x, y)/f(x)$  is the conditional density of  $Y$  given  $X = x$  (when  $f(x) > 0$ ), and the first equation in (6) gives

$$g(y | x) = g(y | x \cup \{u\}) \quad \text{whenever} \quad f(x \cup \{u\}) > 0,$$

meaning that  $X$  and  $Y$  are independent and  $g(y | x) = g(y)$  does not depend on  $x$ . The second equation in (6) is then equivalent to

$$g(y)(\beta(u) - \lambda(x, u)) = g(y \cup \{u\}),$$

so  $\lambda(x, u) = \lambda(u)$  does not depend on  $x$ . Consequently, by induction,

$$f(x) \propto \prod_{u \in x} \lambda(u), \quad g(y) \propto \prod_{u \in y} (\beta(u) - \lambda(u)),$$

whereby also the ‘only if’ part is verified.

**Example 1.** Consider the very simple case with  $\lambda(\emptyset, u) = \beta > 0$  and  $\lambda(x, u) = 0$  whenever  $n(x) > 0$ , that is,

$$f(\emptyset) = \frac{e^{|S|}}{1 + b}, \quad f(\{u\}) = \frac{\beta e^{|S|}}{1 + b}, \quad f(x) = 0 \quad \text{whenever} \quad n(x) > 1,$$

meaning that  $n(X) \leq 1$  and, with probability  $b/(1 + b)$ ,  $n(X) = 1$ , in which case  $X$  consists of a uniformly distributed point in  $S$ . Note that  $b = \beta|S|$ , and, by Algorithm 1 and Theorem 1, conditional on  $X = x$ , the points in  $Y(x)$  are independent and uniformly distributed in  $S$ . Thus, conditional on  $(n(X), n(Y)) = (m, n)$ , the  $m + n$  points in  $X$  and  $Y$  are independent and uniformly distributed in  $S$ . So the joint distribution of  $X$  and  $Y$  is effectively given by the distribution of  $(n(X), n(Y))$ . Defining

$$\pi_{m,n} = P(n(X) = m, n(Y) = n), \quad m, n \in \{0, 1, \dots\},$$

we have, for any  $(x, y)$  with  $(n(x), n(y)) = (m, n)$ ,

$$\pi_{m,n} = \frac{|S|^{m+n} e^{-2|S|}}{m!n!} \pi(x, y)$$

and  $\pi_{m,n} = 0$  if  $m \geq 2$ . Hence, (5) is seen to be equivalent to

$$\pi_{0,0} = e^{-b}, \quad \pi_{1,n} = \frac{b^{n+1}}{(n+1)!} e^{-b} - \pi_{0,n+1}, \quad (b+n)\pi_{0,n} = \pi_{1,n} + (n+1)\pi_{0,n+1},$$

where the two first equations follow from the fact that  $n(X) + n(Y)$  is Poisson distributed with parameter  $b$ , and the last equation follows since (5) gives

$$(b+n) \frac{0!n!}{|S|^n e^{2|S|}} \pi_{0,n} = 0 + 0 + |S| \frac{1!n!}{|S|^{n+1} e^{2|S|}} \pi_{1,n} + |S| \frac{0!(n+1)!}{|S|^{n+1} e^{2|S|}} \pi_{0,n+1}.$$

Consequently, the  $\pi_{m,n}$  are determined by  $\pi_{0,1}$ , since  $\pi_{1,n}$  is determined by  $\pi_{0,n+1}$  for  $n = 0, 1, \dots$ , and  $\pi_{0,n+1}$  is determined by  $\pi_{0,1}$  for  $n = 1, 2, \dots$ , since

$$\pi_{0,n+1} = \frac{1}{n} \left[ (b+n)\pi_{0,n} - \frac{b^{n+1}}{(n+1)!} e^{-b} \right], \quad n = 1, 2, \dots \tag{18}$$

Using induction, it follows easily from (18) that

$$\pi_{0,n+1} = \left[ \prod_{i=1}^n \frac{b+i}{i} \right] \left[ \pi_{0,1} - e^{-b} \sum_{i=2}^{n+1} \frac{b}{i(i-1)} \prod_{j=1}^{i-1} \frac{b}{b+j} \right], \quad n = 1, 2, \dots, \tag{19}$$

which can be rewritten as

$$\begin{aligned} \pi_{0,n+1} &= \frac{\Gamma(b+n+1)}{\Gamma(n+1)\Gamma(b+1)} \\ &\times \left[ \pi_{0,1} - b^{-b} \left( \Gamma(b+1, b) - \frac{\Gamma(b+n+1, b)\Gamma(b+1)}{\Gamma(b+n+1)} \right) \right. \\ &\quad \left. - e^{-b} b \left( 1 - \frac{b^n \Gamma(b+1)}{(n+1)\Gamma(b+n+1)} \right) \right], \quad n = 1, 2, \dots, \tag{20} \end{aligned}$$

where  $\Gamma(a, x) = \int_x^\infty t^{a-1} e^{-t} dt$  is the incomplete gamma function which has the property that  $\Gamma(a, x) = (a-1)\Gamma(a-1, x) + x^{a-1} e^{-x}$ .

As  $n \rightarrow \infty$ , we have  $\pi_{0,n+1} \rightarrow 0$ . Since  $\prod_{i=1}^n (b+i)/i \rightarrow \infty$  as  $n \rightarrow \infty$ , (19) implies that

$$\pi_{0,1} = e^{-b} \sum_{i=2}^\infty \frac{b^i}{i(i-1)} \frac{\Gamma(b+1)}{\Gamma(b+i)}.$$

Using a similar argument, but taking (20) as the starting point, we obtain

$$\pi_{0,1} = b^{-b} (\Gamma(b+1, b) - \Gamma(b+1)) + e^{-b} b.$$

Inserting this into (20) and noting that

$$P(n(Y) = n) = \pi_{0,n} + \pi_{1,n} = \pi_{0,n} + \frac{b^{n+1}}{(n+1)!} e^{-b} - \pi_{0,n+1},$$

we find that the marginal distribution of  $n(Y)$  is given by

$$P(n(Y) = n) = \frac{b^{1-b}(\Gamma(b+n) - \Gamma(b+n, b))}{\Gamma(n+1)}, \quad n = 1, 2, \dots$$

**Example 2.** The case above extends to when  $\lambda(x, u) = \mu_{n(x)}$  depends only on the number of points in  $x$ , where, for  $n = 0, 1, \dots$ ,  $0 \leq \mu_n \leq \beta$  and if  $\mu_n = 0$  then  $\mu_{n+1} = 0$ . Again, conditional on  $(n(X), n(Y)) = (m, n)$ , the  $m+n$  points in  $X$  and  $Y$  are independent and uniformly distributed in  $S$ , and solving (5) becomes equivalent to solving

$$(b+m+n)\pi_{m,n} = \lambda_{m-1}\pi_{m-1,n} + (b-\lambda_m)\pi_{m,n-1} + (m+1)\pi_{m+1,n} + (n+1)\pi_{m,n+1}$$

for  $m, n = 1, 2, \dots$ , where  $\lambda_m = |S|\mu_m$ ,  $\pi_{-1,n} = 0$ , and  $\pi_{m,-1} = 0$ . It follows by induction that the  $\pi_{0,n}$ ,  $n = 0, 1, \dots$ , determine all the  $\pi_{m,n}$ ,  $m, n = 1, 2, \dots$ . We know that  $\pi_{0,0} = e^{-b}$  but, in general, for  $n \geq 1$ , we do not have a simple recursion for the  $\pi_{0,n}$ ; this is in contrast to (18). So finding an expression for  $\pi_{0,n}$  when  $n \geq 1$  seems now to be a much harder problem.

In conclusion, apart from the rather trivial case where  $X$  and  $Y$  are independent Poisson processes (see Proposition 3), the joint distribution of  $X$  and  $Y$  seems to be complicated. Furthermore, apart from simple cases (such as Example 1), the marginal distribution of  $Y$  also seems to be very complicated.

### Acknowledgements

We thank Ege Rubak and Frederic R. Paik Schoenberg for helpful discussions. Constructive comments from the anonymous referees are appreciated. Supported by the Danish Natural Science Research Council, grant numbers 272-06-0442 and 09-072331, ‘Point process modelling and statistical inference’, and by the Centre for Stochastic Geometry and Advanced Bioimaging, funded by a grant from the Villum Foundation.

### References

- [1] BADDELEY, A. AND TURNER, R. (2005). Spatstat: an R package for analyzing spatial point patterns. *J. Statist. Software* **12**, 1–42.
- [2] BADDELEY, A. AND TURNER, R. (2006). Modelling spatial point patterns in R. In *Case Studies in Spatial Point Process Modeling* (Lecture Notes Statist. **185**), eds A. Baddeley *et al.*, Springer, New York, pp. 23–74.
- [3] BERTHELSEN, K. K. AND MØLLER, J. (2002). A primer on perfect simulation for spatial point processes. *Bull. Brazilian Math. Soc.* **33**, 351–367.
- [4] BESAG, J. (1977). Some methods of statistical analysis for spatial data. *Bull. Internat. Statist. Inst.* **47**, 77–91.
- [5] CARTER, D. S. AND PRENTER, P. M. (1972). Exponential spaces and counting processes. *Z. Wahrscheinlichkeitsth.* **21**, 1–19.
- [6] FERRARI, P. A., FERNÁNDEZ, R. AND GARCIA, N. L. (2002). Perfect simulation for interacting point processes, loss networks and Ising models. *Stoch. Process. Appl.* **102**, 63–88.
- [7] FIKSEL, T. (1984). Simple spatial-temporal models for sequences of geological events. *Elektron. Informationsverarb. Kybernet.* **20**, 480–487.
- [8] GEYER, C. (1999). Likelihood inference for spatial point processes. In *Stochastic Geometry* (Monogr. Statist. Appl. Prob. **80**), eds O. E. Barndorff-Nielsen, W. S. Kendall, and M. N. M. van Lieshout, Chapman and Hall/CRC, Boca Raton, FL, pp. 79–140.
- [9] ILLIAN, J., PENTTINEN, A., STOYAN, H. AND STOYAN, D. (2008). *Statistical Analysis and Modelling of Spatial Point Patterns*. John Wiley, Chichester.
- [10] KELLY, F. P. AND RIPLEY, B. D. (1976). A note on Strauss’ model for clustering. *Biometrika* **63**, 357–360.
- [11] KENDALL, W. S. (1998). Perfect simulation for the area-interaction point process. In *Probability Towards 2000* (Lecture Notes Statist. **128**), eds L. Accardi and C. C. Heyde, Springer, New York, pp. 218–234.
- [12] KENDALL, W. S. AND MØLLER, J. (2000). Perfect simulation using dominating processes on ordered spaces, with application to locally stable point processes. *Adv. Appl. Prob.* **32**, 844–865.

- [13] MØLLER, J. (1989). On the rate of convergence of spatial birth-and-death processes. *Ann. Inst. Statist. Math.* **41**, 565–581.
- [14] MØLLER, J. AND SCHOENBERG, R. P. (2010). Thinning spatial point processes into Poisson processes. *Adv. Appl. Prob.* **42**, 347–358.
- [15] MØLLER, J. AND SØRENSEN, M. (1994). Parametric models of spatial birth-and-death processes with a view to modelling linear dune fields. *Scand. J. Statist.* **21**, 1–19.
- [16] MØLLER, J. AND WAAGEPETERSEN, R. P. (2004). *Statistical Inference and Simulation for Spatial Point Processes*. Chapman and Hall/CRC, Boca Raton, FL.
- [17] PRESTON, C. J. (1977). Spatial birth-and-death processes. *Bull. Internat. Statist. Inst.* **46**, 371–391.
- [18] PROPP, J. G. AND WILSON, D. B. (1996). Exact sampling with coupled Markov chains and applications to statistical mechanics. *Random Structures Algorithms* **9**, 223–252.
- [19] RIPLEY, B. D. (1977). Modelling spatial patterns (with discussion). *J. R. Statist. Soc. B* **39**, 172–212.
- [20] RIPLEY, B. D. AND KELLY, F. P. (1977). Markov point processes. *J. London Math. Soc.* **15**, 188–192.
- [21] STEPHENS, M. (2000). Bayesian analysis of mixture models with an unknown number of components—an alternative to reversible jump methods. *Ann. Statist.* **28**, 40–74.
- [22] STOYAN, D., KENDALL, W. S. AND MECKE, J. (1995). *Stochastic Geometry and Its Applications*, 2nd edn. John Wiley, Chichester.
- [23] STRAUSS, D. J. (1975). A model for clustering. *Biometrika* **62**, 467–475.