POISSON

A program for spatial point generation using Poisson processes

Department of Mining and Mineral Engineering
The University of Leeds
September 2002

1. Introduction

The purpose of the program is to generate spatial points in R^2 as realisations of user-defined Poisson processes. The type of processes implemented by the program are: homogeneous Poisson process, non-homogeneous Poisson process, Poisson cluster process, simple Cox process and simple Boolean model.

The realisation is to be constructed in a rectangular area, $A=[x_1,x_2]\times[y_1,y_2]$, where x_1, x_2, y_1 and y_2 can be specified. A is divided into quadrate cells with each individual cell being treated as an all bounded Borel set. The number of cells in X and Y can also be specified by the user of the program. For this arrangement, all cells are disjoint and therefore the simulation of Poisson points for each cell can be done independently.

The commands to be used are all listed under the master menu of "SpatialPattern". Select the corresponding command will bring up the related parameter specification window. Closing the window will activate the simulation and the generated points will be displayed once the process finishes.

2. Homogeneous Poisson process

Homogeneous process is the simplest possible stochastic mechanism representing the spatial point patterns. It generates realisations of stationary and isotropic characteristics which resemble complete spatial randomness (CSR) in applications. The process defines a spatial points pattern which bears the following properties (characteristics):

- For any bounded Borel set A in the region of consideration $\Re$, the number of points falling inside A, $N(A)$, follows a Poisson distribution with mean $\lambda\cdot\nu(A)$, where $\nu(A)$ is the measure of A. λ is referred to as the density of the distribution. In two-dimensional case, A is a planar sub-region of $\Re$ and $\nu(A)$ is the area of the sub-region.
- Conditioned on $N(A) = n$, the n event points are uniformly distributed inside the sub-region A. In two-dimensional case, the events follow bi-variate uniform distribution.
- For any disjoint sub-regions $B_1, B_2, \ldots, B_k$ within the region $\Re$, $N(B_1), N(B_2), \ldots, N(B_k)$ are independent random variables.

Two distinct ways can be used to generate realisations of homogeneous Poisson process. One is to use the property that the spatial differences (areas in 2D case) between successive point events follow an exponential distribution with parameter λ (the density of the Poisson process). Different implementations can be derived to explore this property in 2D and 3D cases. The other way is to simulate the Poisson variable $N(A)$ directly. Simulation of $N(A)$ may sometimes be time-consuming, the implementation nevertheless is simpler and less error prone. In the current exercise, direct Poisson variable simulation is used.

The procedures for the realisation of homogeneous Poisson process in a two dimensional region $\Re$ are given as:

- 1). Sub-divide the region $\Re$ into m sub-regions, $A_1, A_2, \ldots, A_m$. In most of the cases, region $\Re$ is a rectangle and $A_1, A_2, \ldots, A_m$ are disjoint rectangular

quadrates with $\Re = \bigcup_i A_i$. Work out the mean of the Poisson distribution for each quadrate as $m_i = l_i \cdot n(A_i)$, where l_i is the density for quadrate *i*. For homogeneous case, $l_1 = l_2 = \ldots = l_m = l$.

- 2). For each of the *m* sub-regions A_i, generate a random variable N_i based on the Poisson distribution function:

$$P(N_i = n) = e^{-m_i} \frac{m_i^n}{n!}$$

- 3). For each of the *n* events for quadrate A_i, generate two random variables according to the uniform distribution and use them as the locations inside the sub-region.
- 4). Repeat procedures 2 and 3 until all quadrates in $\Re$ are visited.

In step 2, the simulation of Poisson variable is from Ross [5]. To simulate a Poisson random variable with mean *m*, generate a sequence of independent random variables uniformly distributed within (0,1), U_1, U_2, …, stopping when the following condition is reached:

$$\prod_{i=1}^{k} U_i < e^{-m}$$

then the variable $n=k$ has the desired distribution. An alternative to the above condition is by taking the logarithms and the condition becomes:

$$\sum_{i=1}^{k} \log(U_i) < -m \quad or \quad \sum_{i=1}^{k} -\log(U_i) > m$$

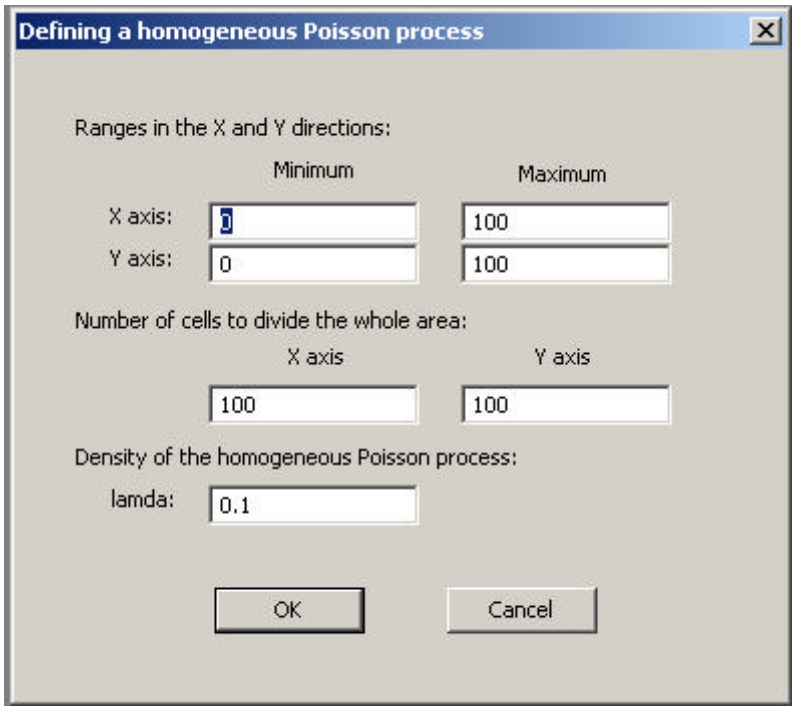


Figure 1 Parameters for homogeneous Poisson process

Based on our experience, the alternative form is preferred although it takes longer computation time due to the logarithmic transformation involved in each step of the

simulation. The reason is simple. When $m \to \infty$, $e^{-m} \to 0$, so for large m value, e^{-m} is small. As $U_i \in (0,1)$ so a large number of uniform variables U_i are needed to reach the condition. Computers are always problematic when dealing with small numbers and therefore as k increases, the rounding error soon renders $\prod U_i$ inaccurate or even meaningless. By using logarithms, this rounding error problem is avoided.

Figure 1 is the dialogue window for specifying the parameters to be used to generate a realisation of a homogeneous Poisson process. The region of simulation $\Re$ is a rectangle and $\Re$ is sub-divided into rectangular sub-regions by specifying the number of cells required in both axis directions. Density λ is specified directly by typing the value in the data box provided.

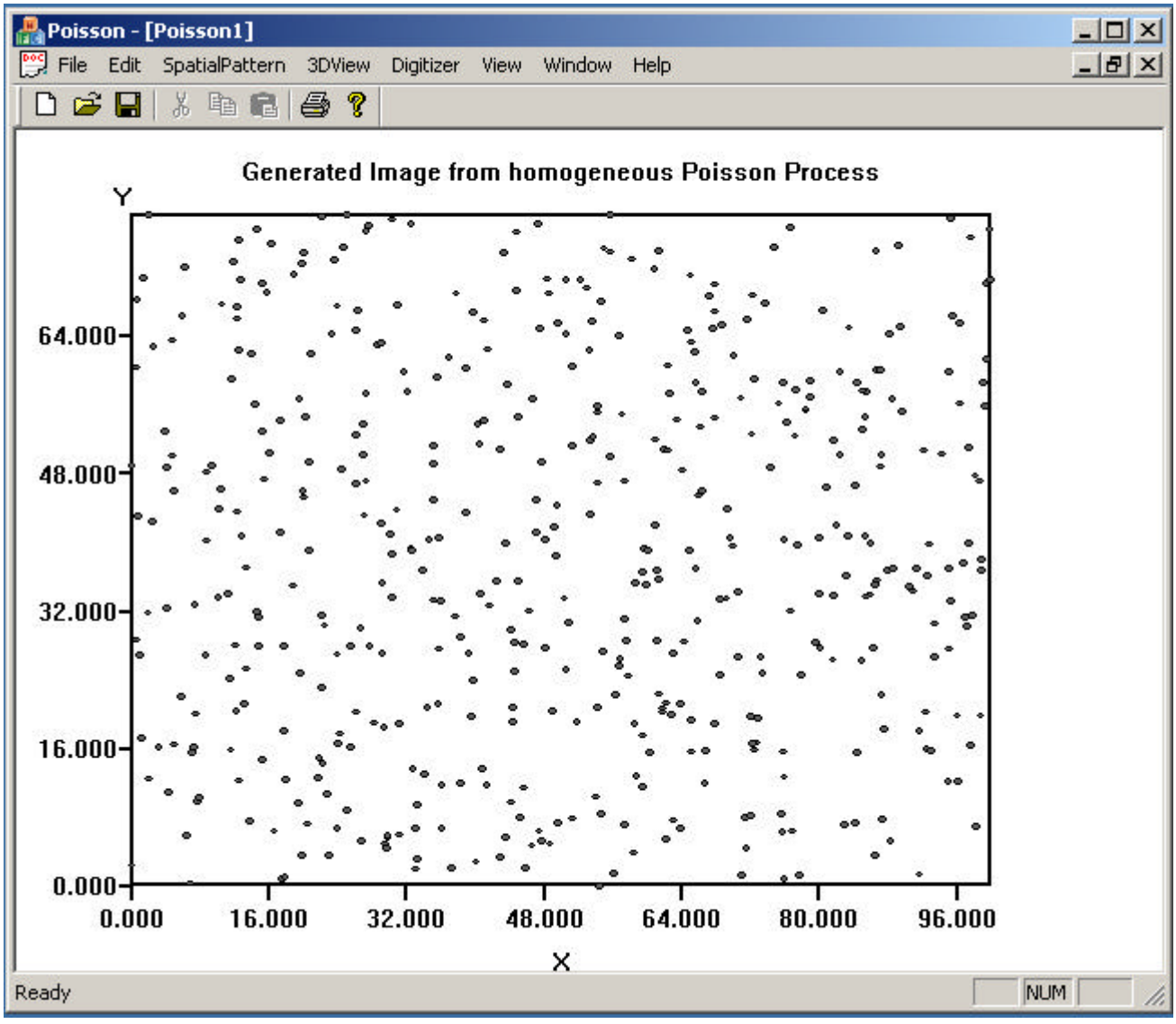


Figure 2 Example of a realisation of a homogeneous Poisson process

Figure 2 shows an example of a realisation of a homogeneous Poisson process defined in Figure 1. Homogeneity of the point pattern is obvious just by way of visual inspection.

QUESTION: How can we control the number of events to be an exact value?

3. Non-homogeneous Poisson process

Non-homogeneous Poisson process represent a class of point patterns with non-stationary characteristics. The only difference of this process to the homogeneous counterpart is that the density l within the region of consideration $\Re$ is no longer constant, it is replaced by a location dependent density function $l(X)$.

Similar to homogeneous Poisson process, non-homogeneous process has the following properties:

- For any bounded Borel set A in the region of consideration $\Re$, the number of points falling inside A, $N(A)$, follows a Poisson distribution with mean $\int_A l(X) \cdot dX$, where X is location variable within A.
- Conditioned on $N(A) = n$, the n event points in A are independently distributed with PDF directly proportional to $l(X)$.
- For any disjoint sub-regions $B_1, B_2, \ldots, B_k$ within the region $\Re$, $N(B_1), N(B_2), \ldots, N(B_k)$ are independent random variables.

Once the density function $l(X)$ is defined, the difficulty in simulating a non-homogeneous Poisson process lies in the second property listed above, i.e., the distribution of n independent events inside A according to the defined $l(X)$. In general $l(X)$ is location dependent and cannot be decomposed into independent functions. For example, in two dimensional case, $l(X)$ depends on x and y and for general consideration $l(X)$ does not decompose into the form $l_1(x) \cdot l_2(y)$. In this case, a direct sampling from the joint PDF for the event locations (x,y) can not be achieved.

A solution to this problem has been described in various sources, for example, see Diggle [1], Lewis and Sedler [4]. Effectively the method proposed is a Poisson thining process. The maximum intensity value l_{max} within the region is found first. An intermediate realisation of a homogeneous Poisson process with intensity value equal to l_{max} is simulated followed by a thining process to obtain the final image. The thining process basically loops through all points generated in the intermediate realisation and retain or discard the event at location X based on the probability $l(X)/l_{max}$.

The procedures for the realisation of non-homogeneous Poisson process in a two dimensional region $\Re$ are given as:

- 1). According to the defined density function $l(X)$, find the maximum density value l_{max} within the simulation region $\Re$. This is normally a simple functional maxima problem and various analytical and numerical methods can be used.
- 2). Simulate a homogeneous Poisson process within $\Re$ with density value equal to l_{max}, following the procedures described in section 2. The total number of events generated is N.
- 3). For each event i ($i=1, 2, \ldots, N$) generated in the last step, generate a probability P_i from an uniform distribution U(0,1) and retain/discard the event for the final image according the following relation:

$$\begin{cases} \text{if } \dfrac{l(X_i)}{l_{\max}} > P_i & \text{event } i \text{ is retained} \\ \text{else} & \text{event } i \text{ is discarded} \end{cases}$$

where X_i is the location of event i.

- 4). Repeat procedures 3 until all points generated in step 2 are processed.

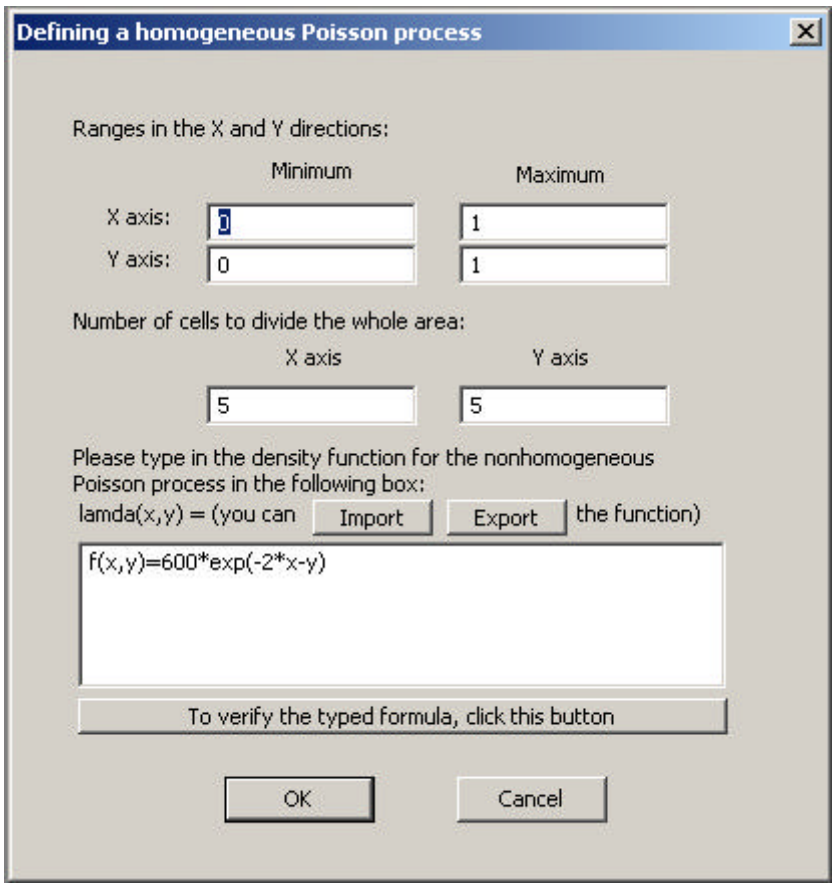


Figure 3 Parameters for non-homogeneous Poisson process

Figure 3 is the dialogue window for generation the realisation of a non-homogeneous Poisson process. For parameter specification, the only difference to homogeneous process is the definition of the density function $l(X)$. In the program, the function is typed in by a text editor and the program will parse through automatically the typed equation to work out the variables and function to be used. If any syntax error (such as unbalanced parentheses) is detected, a warning message will be given. The program however can not detect or correct any errors in the equation itself.

As an example, the density function of $l(\boldsymbol{x}, \boldsymbol{y}) = 600 \cdot \boldsymbol{e}^{(-2x-y)}$ is typed as *f(x,y)*=600**exp*(-2**x-y*) in the text editor. Some other functions supported by the program so far are listed in the following table:

Table 1 Functions supported by the program so far

Functions	Mathematical form	Typed form in text editor
Absolute	$\|x\|$	*abs(x)*
Exponential	e^x	*exp(x)*

Square root	$\sqrt{x}$	*sqrt(x)*
Sine	*sin(x)*	*sin(x)*
Cosine	*cos(x)*	*cos(x)*
Tangent	*tan(x)*	*tan(x)*
Factorial	*n!*	*fact(n)*
Gamma	*g(x)*	*gama(x)*

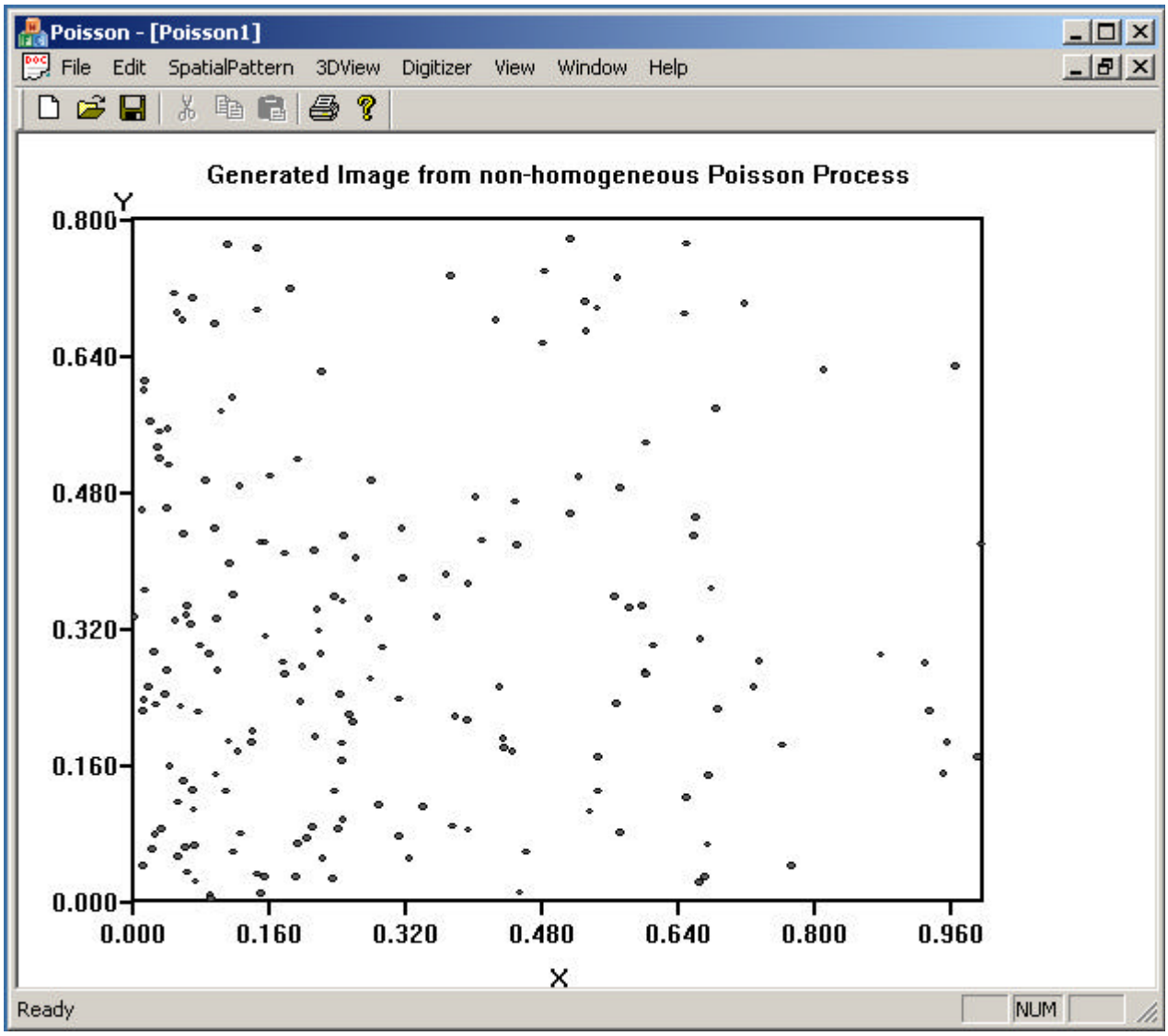


Figure 4 A realisation of the non-homogeneous Poisson process defined in Figure 3

Figure 4 shows the final realisation of the Poisson process with density function defined in Figure 3. This example is also used by Diggle [1]. As another example, Figure 5 shows a nonhomogeneous process with density function defined by:
$l(x, y) = 50 * (1 - \sin(x) \cdot \sin(y))$.

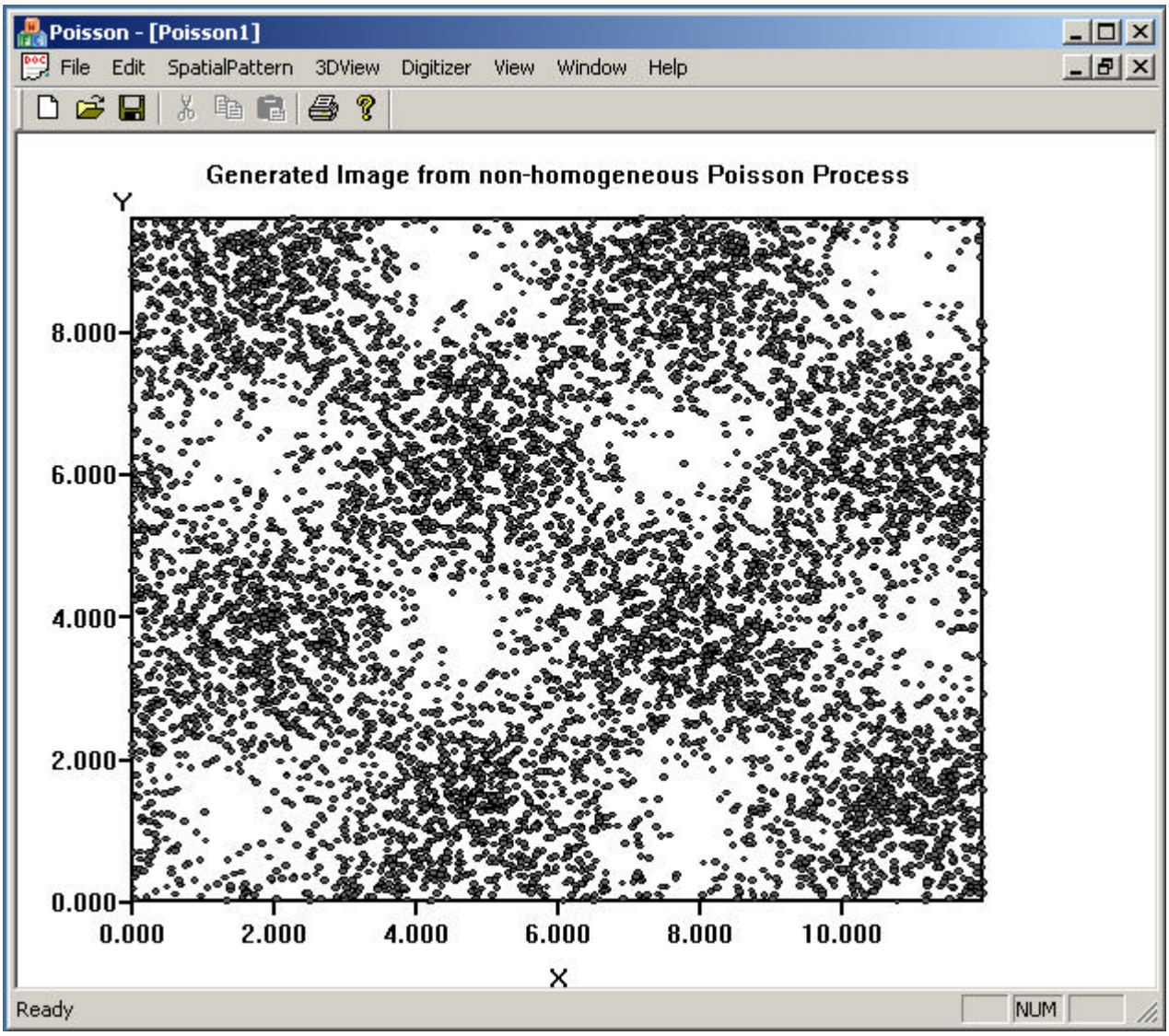


Figure 5 A realisation of a Poisson process with density function

$$l(x, y) = 50 * (1 - \sin(x) \cdot \sin(y))$$

4. Poisson cluster process

Poisson cluster process provides a tool for modelling a class of spatial point patterns with aggregated characteristics, or spatial clustering. It is also often referred to as parent-daughter modelling in practice, or hierarchy modelling in which case parent-daughter relations can have multiple generations, such as the one referred in Lee and Einstein [3]. Poisson cluster process actually consists of two separate processes, the parent process and the following daughter process. The final point pattern is normally composed of offspring (daughters) only even though parents are included in some special cases to avoid clusters of zero size of offspring [1].

The realisation of Poisson cluster process will incorporate the following three characteristics:

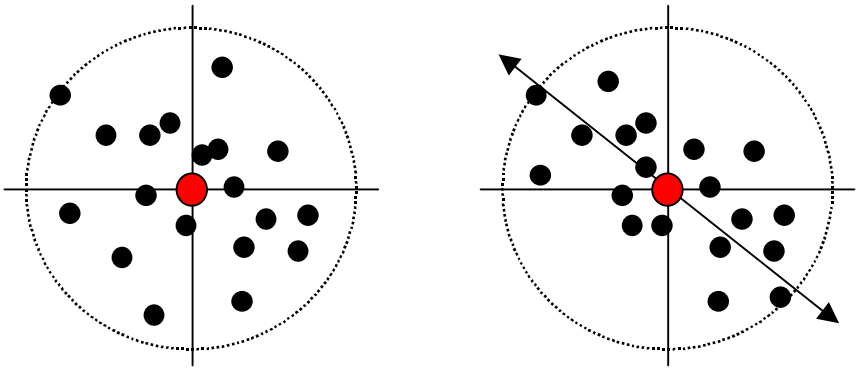
- Parent events are generated from a Poisson process. The process can be homogeneous Poisson process with constant density l, non-homogeneous Poisson process with density function $l(X)$, a doubly-stochastic (Cox) Poisson

process, or even another cluster process in which case a multiple generation parent-daughter model is being constructed. In Poisson cluster modelling, this is referred to as the parent process and it lays the foundation for any subsequent daughter process, which is the combination of the following two definitions. N number of parents are obtained in this step.

- For each parent p_i ($i=1, 2, \ldots, N$), a random number of daughters, d_i ($i=1, 2, \ldots, N$) are produced. d_i is generated independently and identically for each parent from the same probability distribution.
- For each parent p_i, its daughters locations in relative to its position are identically distributed according to the same probability distribution function, $l(\chi)$. In general daughters together form a cluster centred on their parents. The daughter location distributions are also independent in the sense that both daughter points are independently distributed of each other and clusters of daughters from different parent do not interfere with each other. $l(\chi)$ depends on the dimensions being considered. In 2D case, it is a bivariate function, i.e., $l(\chi)=l(x,y)$.

Many interesting Poisson cluster process can be created by variations of the last two properties. Some known processes include Neyman-Scott cluster process, which is used to describe the cluster process where daughter points are independently and identically distributed and Matérn cluster process, where the parent process is a homogeneous Poisson process and each parent produces a Poisson number of offspring generated with the same distribution mean, see van Lieshout [8]. For daughter locations distribution, the usual pattern found in the literature is the uniform distribution of daughter points within a ball (in 3D) or a circle (in 2D) of radius r centred at the parent point, or a distribution specified by a given PDF function. Isotropic distribution is normally assumed even though distributions with some preferential anisotropic directions are suggested.

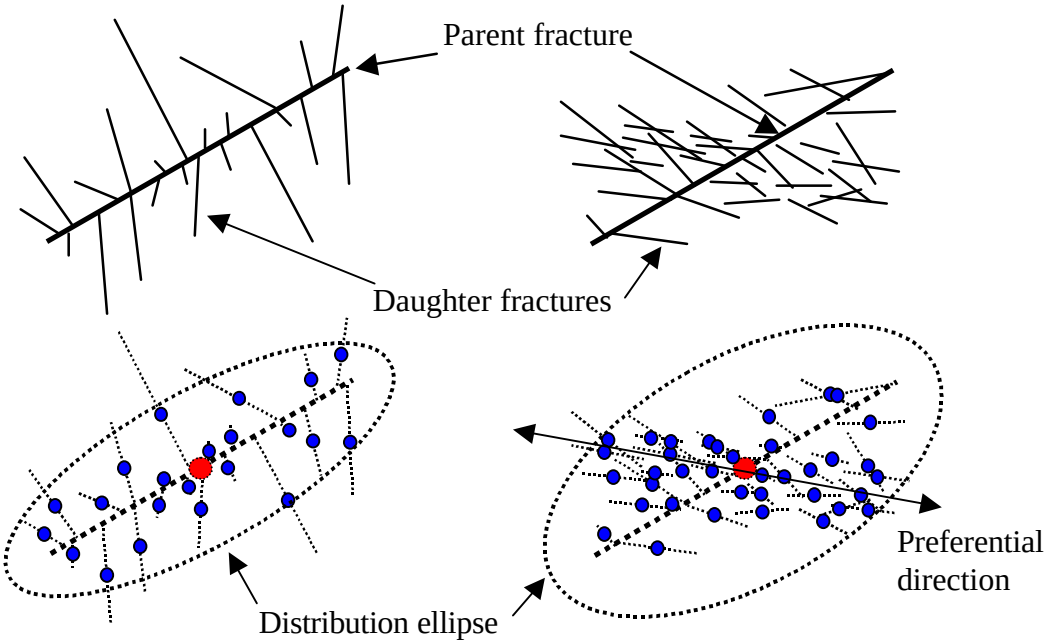
Fractures in rock are signatures of millions of years of geophysical or geothermal activities. These activities can usually be grouped into stages with each stage having a preferential geological movement and hence creating a distinct set of fractures with a preferential orientation. Fractures created in the earlier stages normally tend to have higher possibility of attracting the creation of more new fractures around their vicinity in future geological operations, a situation analogous to parent-daughter model discussed above. The daughter locations distribution for conventional models is normally centred on their parent position, either uniformly within a ball (3D)/a circle (2D), or with a preferential direction, see Figure 6. While for fracture model, daughter locations distribution will be around the whole fracture line and in general with a preferential direction. If we use the centre point to represent parent fracture we can argue that daughters can also be considered to be distributed around the parent point, similar to conventional models. The area and shape of the distribution will, however, be very different. At least an ellipse can only be considered as a realistic representation although a more complicated distribution model can be used, see Figure 7.



Distributed within a circle

Distributed with a preferential direction

Figure 6 Conventional parent-daughter distribution model



Distributed within an ellipse

Distributed with a preferential direction

Figure 7 Special considerations for daughter fracture locations distribution

To generate a realisation of a Poisson cluster process, one needs to generate the parent process first. As mentioned above, parent process can be a homogeneous or non-homogeneous process, or it can even be a cluster process itself if hierarchy structure is being modelled. Once the parent model is ready, one can proceed for daughters' generation. Figure 8 shows the options implemented in the program for the realisation of daughters. For the number of offspring per parent, three options are available: it can be set at a fixed number, be generated from a specified uniform

distribution, or it can be generated from a defined Poisson distribution (hence defining a Metérn process).

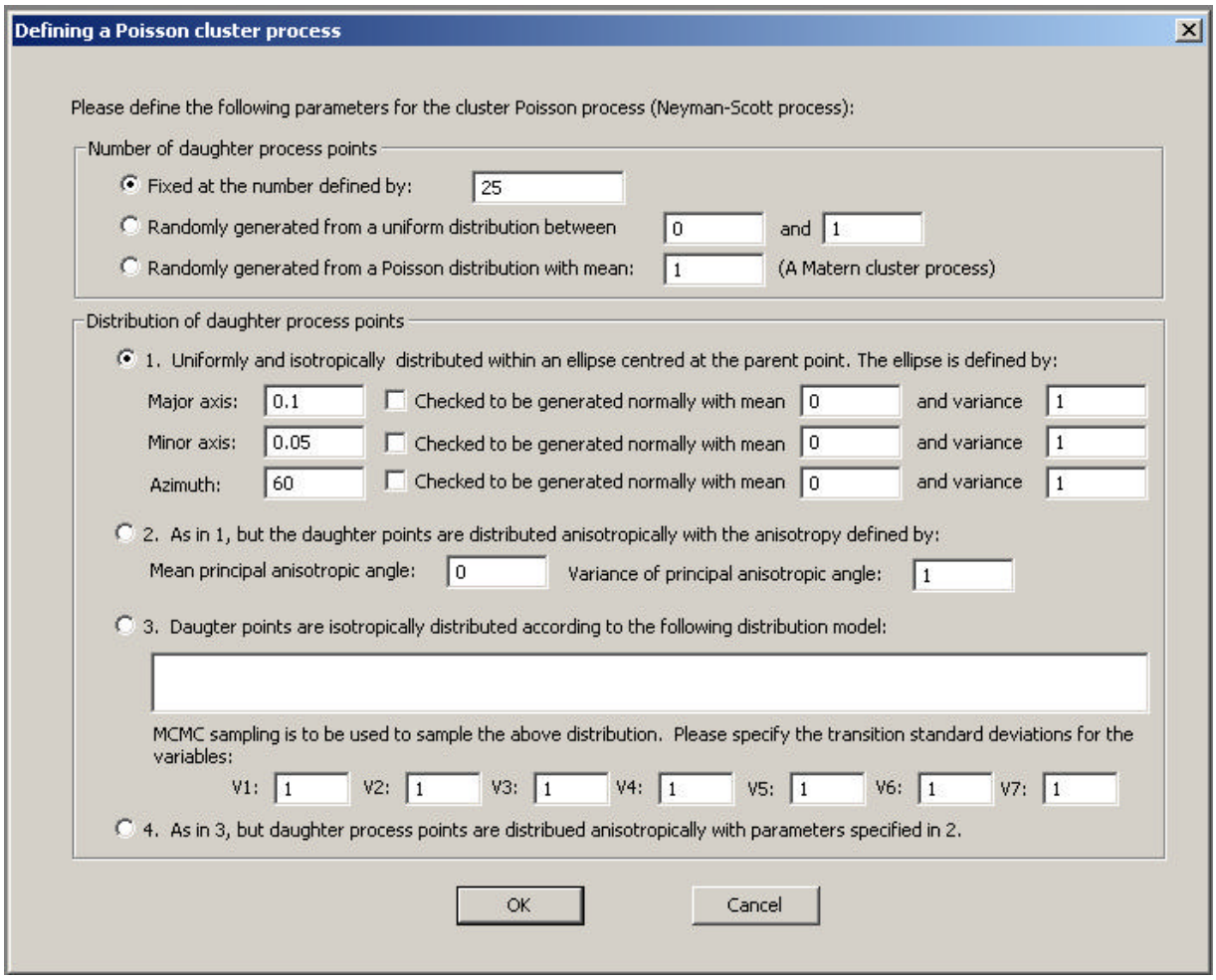


Figure 8 Options for the generation of daughter points of a Poisson cluster process

For distribution of daughters around their parent, two categories of distributions are implemented: uniformly distributed within an ellipse or distributed according a specified PDF function. For each category, a further option of a preferential distribution direction can be specified, see Figure 6 and 7 for detailed illustration.

The distribution ellipse for each parent can be fixed for its axes and major axis azimuth angle, or these attributes can be generated randomly for each parent according to a specified distribution. For fractures simulation, it is more likely to be the later case as parent fractures are likely to have different sizes and orientations, which directly determine the attributes for the distribution ellipse. If the daughter locations are distributed according to a known model, $l(\cdot)$, the model needs only typed into the data box given in option 3 (Figure 8). For example, for a radially symmetrical normal PDF:

$$l(x,y) = \frac{1}{2\pi s^2} e^{-\frac{x^2+y^2}{2s^2}}$$

where x and y are the distances of daughters to their parent. The function is typed in as: *f(x,y)*=(1/(2*3.1415926*0.00625))**exp*(-(*x*x+y*y)/(2*0.00625)), where s^2 is assumed to be 0.00625 (i.e., s=0.025). If more variables are to be included in the PDF, they can be specified as *V** in the formula, where * is a digit between 1 and 9. Note *V1* and *V2* are reserved for variable *x* and *y*. For both categories of distributions, a preferential direction can be specified in which daughters are more likely to be distributed.

When daughter locations distribution is specified by a model, *l(x,y)*, MCMC is used for sampling the distribution for variables *x* and *y*. As *l(·)* normally contains more than two variables (at least *x* and *y*), MCMC will be a more suitable sampling technique here. In the program, the initial values implicitly start at zeros and initial burn-in period are set at 100 internally. These default setting should be sufficient for general simple distribution model, such as the example given above. For the MCMC to work properly, a transition variance for each variable must be specified. These should be compatible to the model *l(·)* specified. Metropolis-Hastings algorithm is used for the MCMC process and for each step all variables are sampled. Whether a set of new samples (the next move), (*x*¢,*y*¢), is accepted depends on the following probability:

$$a\left[(\boldsymbol{x},\boldsymbol{y}),(\boldsymbol{x}',\boldsymbol{y}')\right]=\min\left\{1,\frac{\boldsymbol{L}(\boldsymbol{x}',\boldsymbol{y}')}{\boldsymbol{L}(\boldsymbol{x},\boldsymbol{y})}\right\}$$

where (*x*,*y*) is the current point of the move (current state) and (*x*¢,*y*¢) is generated from the transition PDF conditioned on (*x*,*y*), i.e.,

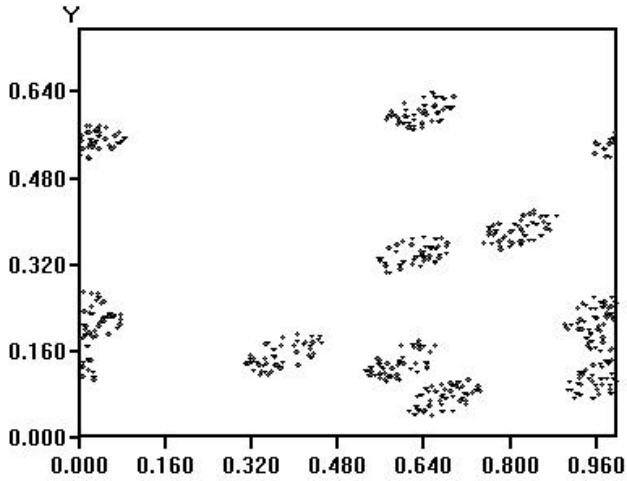
$$(\boldsymbol{x}',\boldsymbol{y}')\sim \boldsymbol{p}((\boldsymbol{X},\boldsymbol{Y})/(\boldsymbol{x},\boldsymbol{y}))$$

and *p* is the proposed transition model. In the program normal PDF is proposed as the transition model. As *p(·)* is symmetric samples are effectively obtained by the random-walk process and therefore all parents can use the same sampling procedure. The likelihood function *L(·)* in MCMC uses the original model *l(·)*, see Sahu [6].

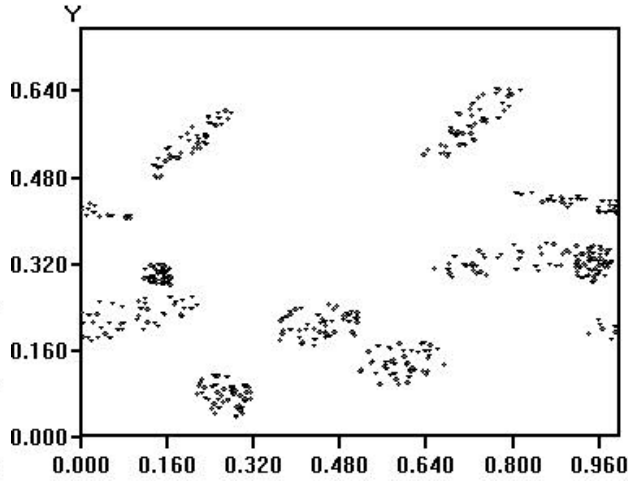
Another problem likely to be encountered during realisation is the edge effect. For example, if the process is to be realised over a rectangular region $\Re$=(*0*,*a*)×(*0*,*b*), some daughters are likely to fall outside the area if the their parents are too close to the side of the rectangle. A simple approach is to discard all points falling outside the rectangle. In order to preserve all the information, however, a better approach is available which basically represents the whole rectangular study region as an edgeless torus by imaging the area is wrapped around a toriod, see Upton [7]. By this approach, the top and bottom edge of the region are considered joined together and so are the left and right edges simultaneously. After this treatment, all daughters generated during the process will be preserved according to the following simple judgements:

$$\begin{cases} \textbf{\textit{In x direction}}: \textbf{\textit{if}}\ \ x<0,\ x=x+a;\ \textbf{\textit{if}}\ \ x>a,\ x=x-a \\ \textbf{\textit{In y direction}}: \textbf{\textit{if}}\ \ y<0,\ y=y+b;\ \textbf{\textit{if}}\ \ y>b,\ y=y-b \end{cases}$$

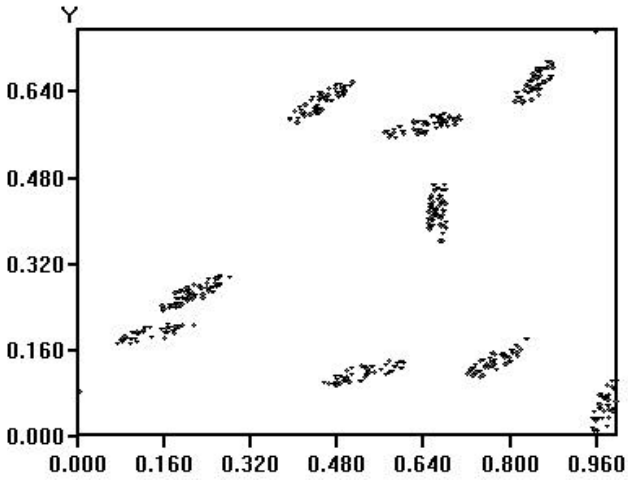
Figure 9 shows a few different realisations of Poisson cluster processes when daughters are distributed within an ellipse centred at their parent. Figure 10 shows some realisations when the daughter locations distribution around their parent are according to a predefined radially symmetrical normal PDF model given as an example above with s^2=0.025.



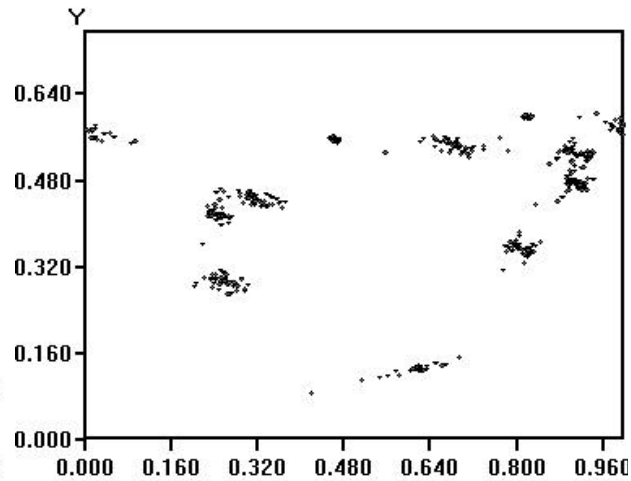
(a). Ellipse: a=0.08, b=0.04, ang=60^o, number of offsprings/parent = 50



(b). Ellipse: a, b and ang are generated randomly, offsprings/parent = 50

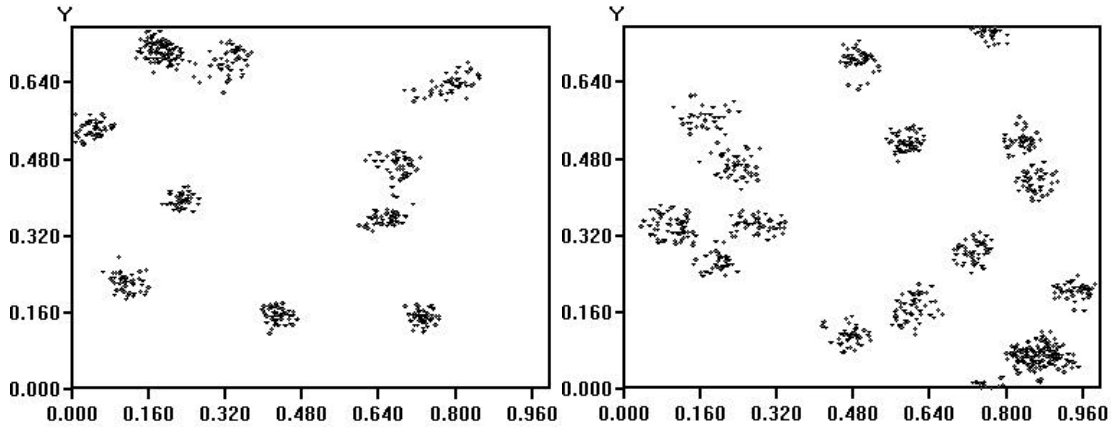


(c). Ellipse: a=0.08, b=0.02, ang generated randomly, Metérn process with mean = 50



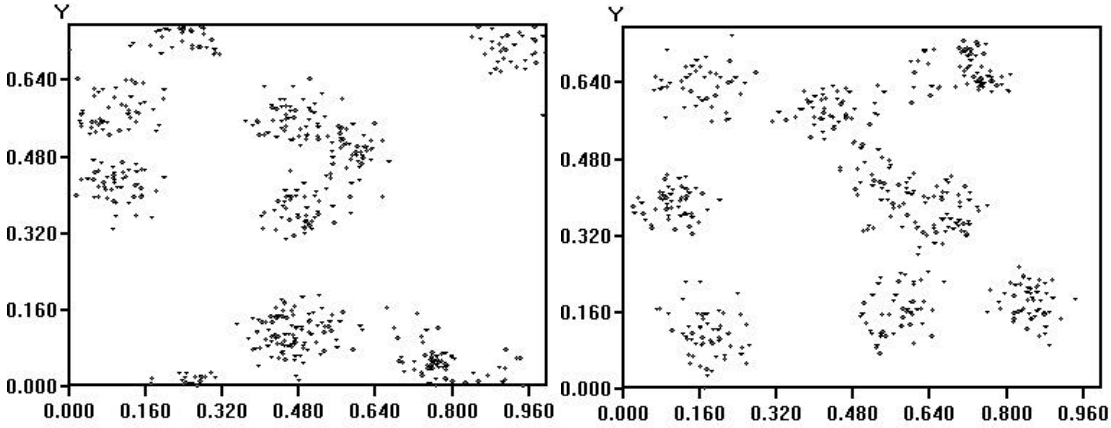
(d). Ellipse: a=0.25, b=0.23, ang=60^o, Metérn process with mean = 50. Preferential distribution direction = 120^o

Figure 9 Some realisation of Poisson cluster process when daughters are uniformly distributed within an ellipse centred at their parents (parents are a realisation of a homogeneous Poisson process with density λ = 10)



(a). Normal PDF distribution model (σ=0.025), transition s.d.v = 0.025, number of offsprings/parent = 50

(b). Normal PDF distribution model (σ=0.025), transition s.d.v. = 0.025, Metérn process with mean = 50



(c). Normal PDF distribution model (σ=0.05), transition s.d.v. = 0.05, number of offsprings/parent = 50

(d). Normal PDF distribution model (σ=0.05), transition s.d.v. = 0.05, Metérn process with mean = 50

Figure 10 Some realisation of Poisson cluster process when daughters are distributed according to a specified normal PDF model (parents are a realisation of a homogeneous Poisson process with density λ = 10)

5. Cox process

Cox process can be viewed as the underlining general model for most of the Poisson processes. Homogeneous, non-homogeneous and cluster Poisson processes can all be treated as special cases of Cox process, see Diggle [1].

In the first sight, Cox process is a logical extension of nonhomogeneous Poisson process. Recall from Section 3 that nonhomogeneous Poisson process is driven by the intensity function $l(X)$ which is and only is location dependent. Cox process takes it a step further and introduce a stochastic element into the intensity function. In other words, the nonhomogeneous Poisson process is now considered to be driven by the underlining intensity function $L(X)$ which is both location dependent and stochastic in nature. It is for this reason Cox process is often referred to as the doubly stochastic Poisson process. Obviously definition of Cox process is done on top of the definition of nonhomogeneous Poisson process (see Section 3). According to Diggle [1], the following two terms define the class of this process:

- $\{ L(X): X \in \mathbb{R}^2 \}$ is a non-negative-valued stochastic process.
- Conditioned on $\{ L(X) = l(X), X \in \mathbb{R}^2 \}$, the events form a nonhomogeneous Poisson process with intensity function $l(X)$.

Kingman [2] established the formal relation between multi-generations Poisson cluster process and Cox process and show that the two classes of processes are equivalent. The formal relation can be expressed as:

$$\Lambda(\boldsymbol{X}) = \sum_{i} m_i(\boldsymbol{G}_n) \cdot \boldsymbol{h}(\boldsymbol{X} - \boldsymbol{X}_i)$$

where G_n are events of the n^{th} generation and $L(X)$ defines a Cox process for the G_{n+1} generation. Obviously $L(X)$ depends on location X and is stochastic as $h(\chi)$ is stochastic in nature.

The equivalence of Cox process and Poisson cluster process gives us the opportunity to model or simulate the cluster process using Cox model. The reverse, however, is not always easy. In other words, to derive a cluster model based on a known Cox process is in general not a simple task.

Cox model can be extremely complex. To demonstrate the idea, a few simple Cox processes are implemented in the program. In principle, Cox process can be simulated in two distinct steps: 1). Simulation of density field $L(X)$ in the region of interest, $\Re$; 2). Simulation of a nonhomogeneous Poisson process with density $l(X) = L(X)$ (see Section 3).

Figure 11 is the dialogue window for generating a realisation of a specified Cox model. The region $\Re$ to be considered for simulation is again a rectangle and the Cox model is specified by a random function. For example, if the Cox model is a exponential PDF with parameter $h(x,y)$ depending on the location (x,y) by the following relation:

$$\boldsymbol{mean} = \frac{1}{h(\boldsymbol{x}, \boldsymbol{y})} = 600 \cdot \boldsymbol{e}^{-2x-y}$$

then the relation is typed in the data box as: f(x,y)=600*exp(-2*x-y), see Figure 11.

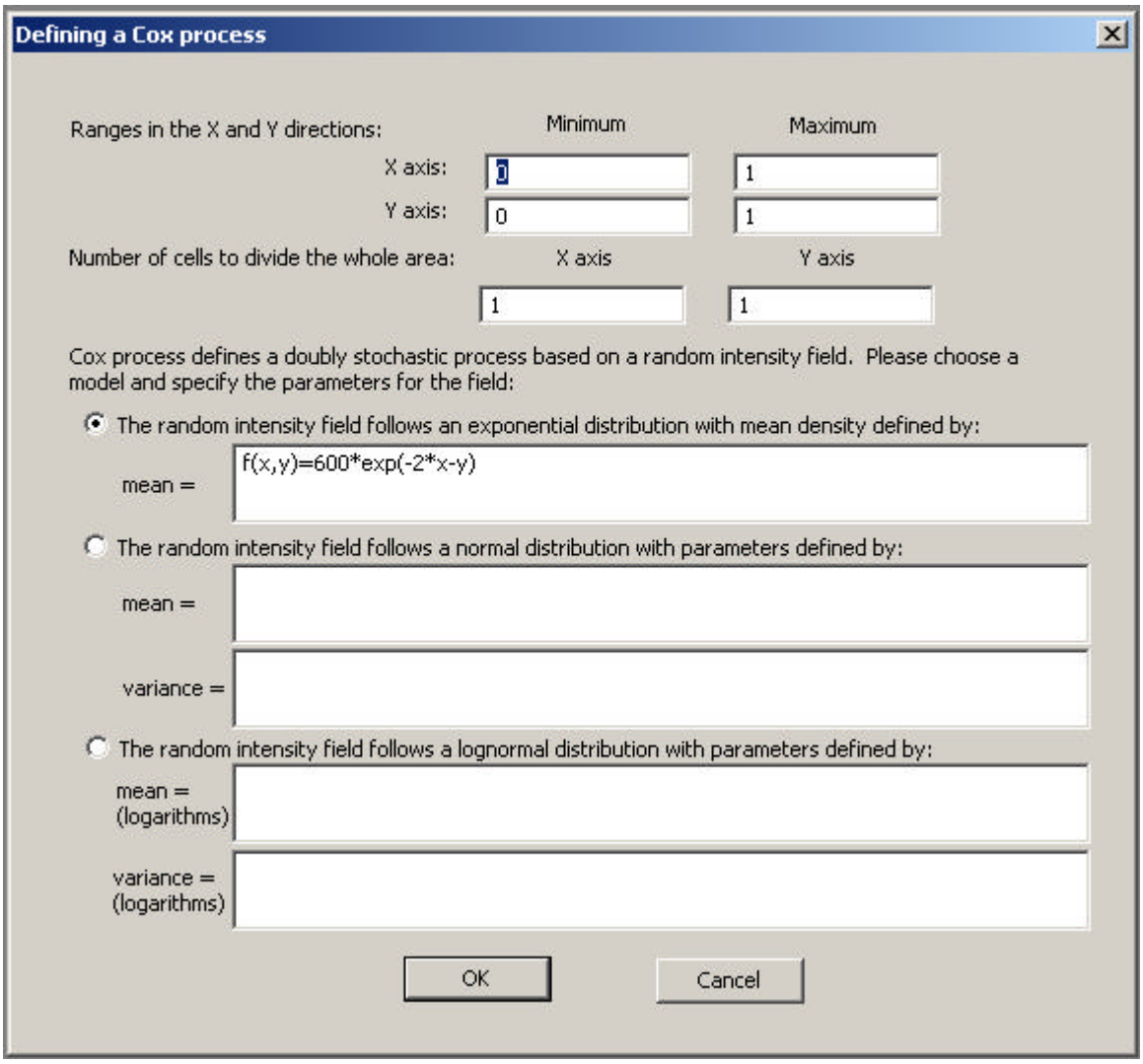


Figure 11 Example of defining a Cox model

Figure 12 is a realisation of the Cox model defined in Figure 11 above. Note the difference between Figure 12 and Figure 4 of a realisation of a nonhomogeneous Poisson process with density function $l(x, y) = 600 * e^{-2x-y}$. These two realisations Are identical in statistical sense (on average), however the thining trends in *x* and *y* directions are less obvious in Figure 12 because of the stochastic element in the density field involved.

Though Cox process is powerful, its modelling in general is not simple. The difficulty lays in finding the effective way to separate the stochastic element, in a single realisation of the random process, from the general deterministic trends of density variations (due to locations) as two of them together composite the final image. An analogous situation in geostatistics is the separation of random noise from fixed trend from a single realisation. A simple approach will be the introduction of assumptions. By assuming the stochastic model or the deterministic trend, the complimentary part can then be derived. This is certainly one of the future research fields to be followed in due cause.

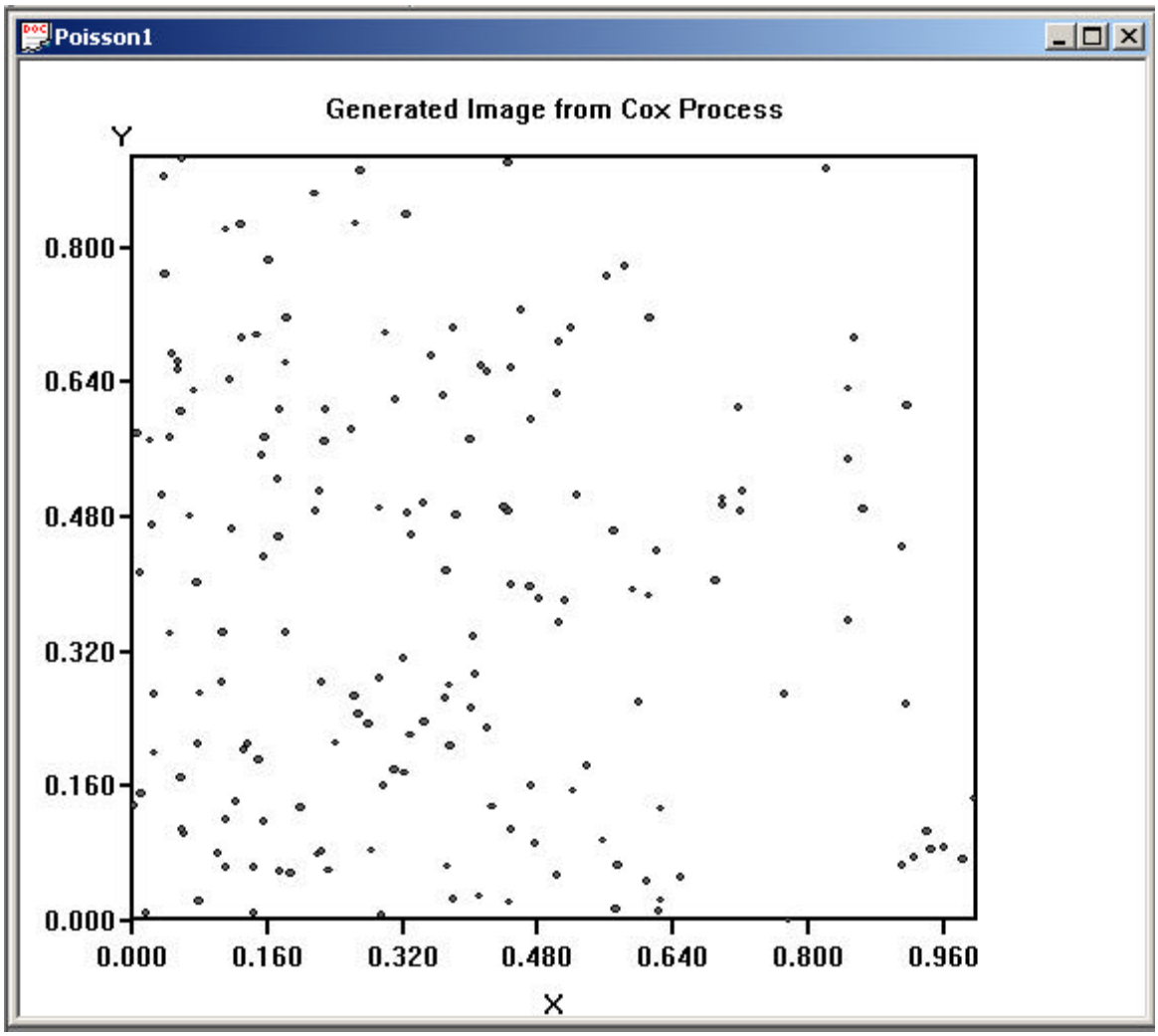


Figure 12 Realisation of a Cox model defined in Figure 11

In Lee and Einstein's paper [3], the Cox model fitted to the dataset is a lognormal stochastic model with mean and variance defined as:

$$\begin{cases} \textbf{\textit{Mean}} = 0.127\ (\textbf{\textit{lognormal scale}}) \\ \textbf{\textit{Variance}} = 0.5 \cdot \boldsymbol{e}^{-2 \cdot \Delta y'} + 0.5 \cdot \boldsymbol{e}^{-3r} \end{cases}$$

where *Dy¢* and *r* are location variables. It is still not clear how the model is derived. Hopefully Lee's research report acquired recently (in microfilms) will reveal some more details.

6. A simple example of a Boolean model

In its simplest form, Boolean model is defined as the union of any independent compact sets generated by an underlying Poisson process [8]. To compose the sets, the points from Poisson process are replaced by "grains", i.e., balls (2D) or circles (2D), centred at the "germs" points, i.e., the generated points, see Cressie [10]. The model is then expressed in terms of the Boolean function:

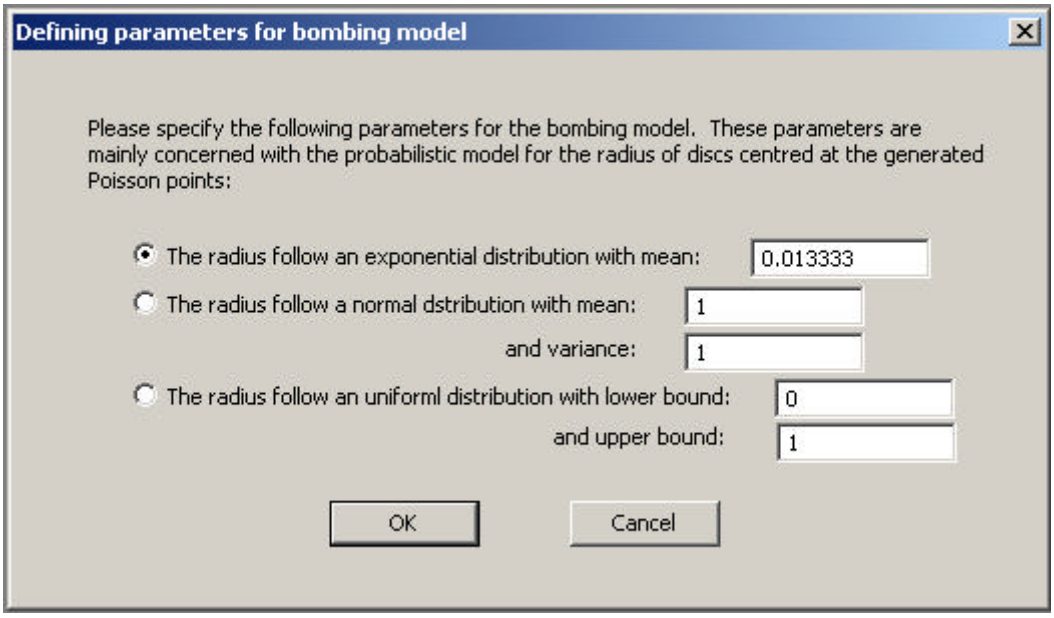


Figure 13 Options for generating realisation of Boolean model

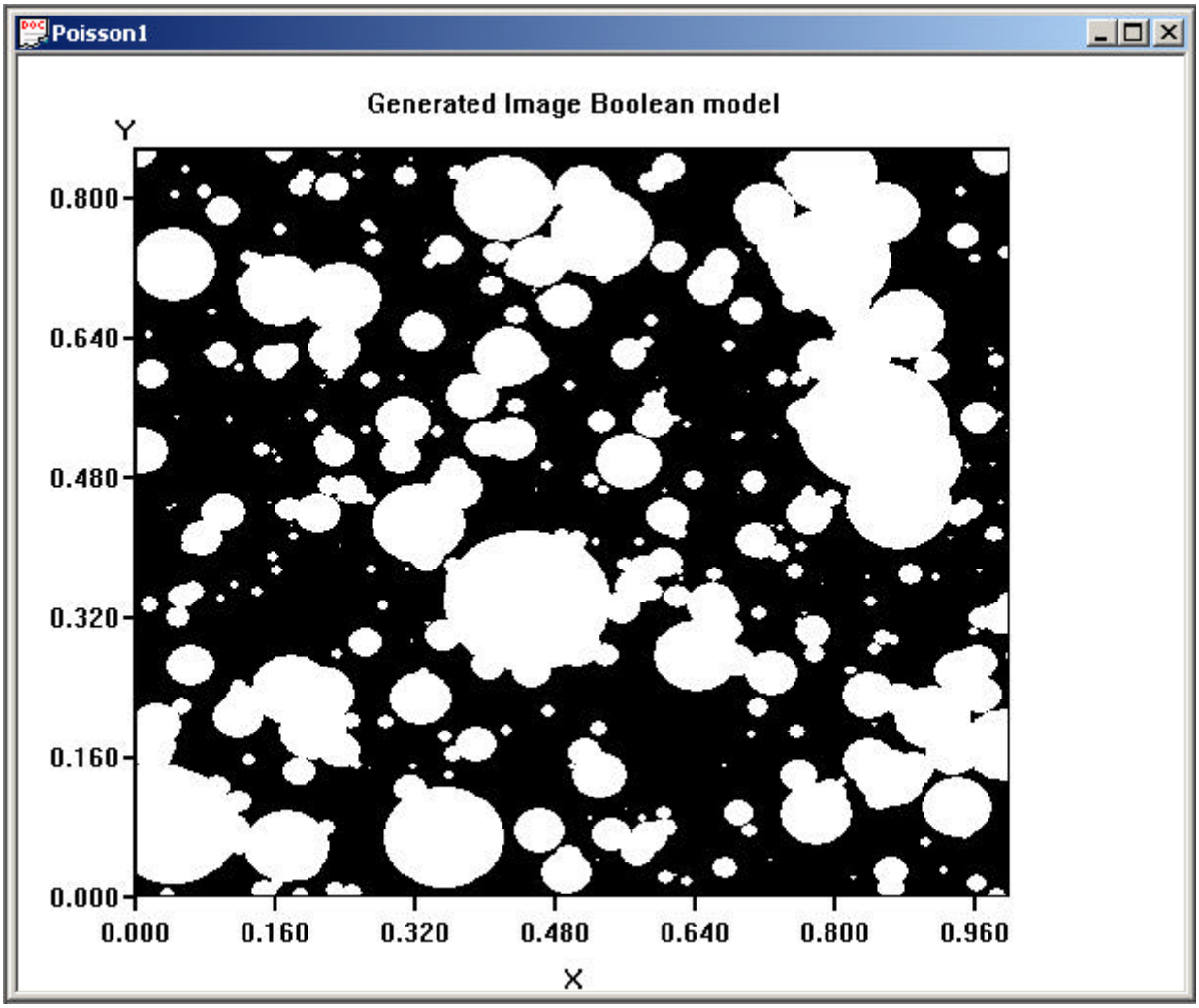


Figure 14 Example of realisation of a Boolean model

$$\boldsymbol{b}(\boldsymbol{X}) = \begin{cases} 1 & \textbf{\textit{if}}\ \ \boldsymbol{d}(\boldsymbol{X}, \boldsymbol{X}_p) \le \boldsymbol{r}_p \quad \textbf{\textit{for}}\ \ \boldsymbol{p} = 1,2,..., \boldsymbol{N} \\ 0 & \textbf{\textit{Otherwise}} \end{cases}$$

where $d(\chi)$ is the distance between point X and a underlying Poisson point X_p, r_p is the corresponding grain size for point p and N is the total number of Poisson points. Figure 13 shows options for generating realisation of a Boolean model and Figure 14 is an example of a generated Boolean model, where the underlying Poisson model is a homogeneous process with density l=500. The radii of circles centred at Poisson points follow an exponential distribution with mean $\frac{1}{75}$ and the region of realisation is a rectangle of dimensions (0, 1) ×(0, 1).

7. References

1. Diggle, P, Statistical analysis of spatial point patterns, Academic press, 1983.
2. Kingman, J., Remarks on the spatial distribution of a reproducing population, Journal of applied probability, vol. 14, 577-583, 1977.
3. Lee, J, Veneziano, D. and Einstein, H., Hierarchical fracture model, 31st US symposium on rock mechanics, A.A.Balkema, 1990.
4. Lewis, P. and Shedler, G., Simulation of non-homogeneous Poisson process by thining, Naval research log quarter, 1979.
5. Ross, S., Introduction to probability models, 6th edition, Academic press, 1997.
6. Sahu, S, Bayesian statistics, course materials downloaded from Dr. S. Sahu's website.
7. Upton, G and Fingleton, B, Spatial data analysis by example, John Wiley & Sons, 1989(?).
8. van Lieshout, M, Markov point processes and their applications, Imperial college press, 2000.
9. Wen, R & Sinding-Larsen, R, Stochastic modelling and simulation of small faults by marked point processes and Kriging, Geostatistics Wollongong '96, Kluwer academic publishers, 1996.
10. Cressie, N., Statistics for spatial data, revised edition, John Wiley & Sons, Inc., 1993.