INSTITUT NATIONAL DE RECHERCHE EN INFORMATIQUE ET EN AUTOMATIQUE

A Polyline Process for Unsupervised Line Network Extraction in Remote Sensing

Caroline Lacoste — Xavier Descombes — Josiane Zerubia

N° 5698

September 2005

Thème COG

Rapport de recherche

N 0249-6399 ISRN INRIA/RR--5698--FR+ENG

A Polyline Process for Unsupervised Line Network Extraction in Remote Sensing

Caroline Lacoste , Xavier Descombes , Josiane Zerubia

Thème COG — Systèmes cognitifs
Projet Ariana

Rapport de recherche n° 5698 — September 2005 — 26 pages

Abstract: This report presents a new stochastic geometry model for unsupervised extraction of line networks (roads, rivers, etc.) from remotely sensed images. The line network in the observed scene is modeled by a polyline process, named CAROLINE. The prior model incorporates strong geometrical and topological constraints through potentials on the polyline shape and interaction potentials. Data properties are taken into account through a data term based on statistical tests. Optimization is done via a simulated annealing scheme using a Reversible Jump Markov Chain Monte Carlo (RJMCMC) algorithm, without any specific initialization. We accelerate the convergence of the algorithm by using appropriate proposal kernels. Experimental results are provided on aerial and satellite images and compared with the results obtained with a previous model, that is a segment process called "Quality Candy".

Key-words: Stochastic geometry, marked point process, simulated annealing, RJMCMC, line network extraction, aerial and satellite images.

Processus de Lignes Brisées pour l'Extraction Non Supervisée de Réseaux Linéiques en télédétection

Résumé : Ce rapport présente un nouveau modèle issu de la géométrie stochastique pour l'extraction non supervisée de réseaux linéiques (routes, rivières, etc.) à partir d'images satellitaires ou aériennes. Le réseau linéique présent dans la scène observée est modélisé par un processus de lignes brisées, appelé CAROLINE. Le modèle *a priori* incorpore de fortes contraintes géométriques et topologiques au travers de potentiels sur la forme des lignes brisées et de potentiels d'interaction. Les propriétés radiométriques sont incorporées via la construction d'un terme d'attache aux données fondé sur des tests statistiques. Un recuit simulé sur un algorithme de type Monte Carlo par Chaîne de Markov (MCMC) à sauts réversibles permet une optimisation globale sur l'espace des configurations d'objets, indépendamment de l'initialisation. L'ajout de perturbations pertinentes permet une accélération de la convergence de l'algorithme. Des résultats expérimentaux obtenus sur des images satellitaires et aériennes sont présentés et comparés à ceux obtenus avec un précédent modèle fondé sur un processus de segments, appelé "Quality Candy".

Mots-clés : Géométrie stochastique, processus ponctuel marqué, recuit simulé, MCMC à sauts réversibles, extraction de réseaux linéiques, images satellitaires et aériennes.

Contents

1 Introduction

Image analysis is an important tool for cartographers to optimize the time spent on ground while improving the accuracy of the final document produced (*i.e.* a map). With the availability of remotely sensed images and advances in computing technologies, many methods have been developed in order to extract cartographic items. In this context, we have been interested in extracting line networks (roads, rivers, etc.) from satellite and aerial images.

Many methods have been developed to tackle this difficult problem, in particular for road network extraction. One possibility is to consider a semi-automatic approach where an operator gives a starting point and a direction that initialize a tracking road algorithm, [28, 9]; some endpoints that may be linked by an algorithm based on dynamic programming [8, 17]; or some checking points that initialize an edge extraction algorithm based on deformable contour models [18], on dynamic programming [12], or on profile analysis [5]. These approaches usually allow a fast and accurate extraction. Nevertheless, the productivity gain of such approaches is weak with respect to the extraction done by an expert.

A second possibility is to consider a fully-automatic approach. The extraction is then an ill-posed problem for which it is difficult to find a good compromise between exhaustivity and specificity. Most automatic methods published in the literature rely on a local optimization process based, for example, on morphological operators [23], on operators dedicated to road extraction [7], on operators based on texture [13], or on neural networks [4]. The major drawback of these low-level techniques is their sensitivity to noise, particularly for high resolution images in which a noise inherent to the observed scene is added (for example, trees shadows on the roads). To reduce this sensitivity to noise, some authors propose to combine different operators [8, 29]. Although they often provide a coarse detection, such techniques are widely used to initialize a network reconstruction procedure. Indeed, they provide road seeds for the initialization of semi-automatic algorithms [2, 30], to construct a graph on which a Markov random field can be defined [25], or to initialize a self-organizing map algorithm [6]. These two-step approaches are, however, strongly sensitive to the pre-detection. For the extraction of thick networks from high resolution images, multi-scale approaches are proposed to reduce the effect of noise while providing an accurate extraction, as in [3, 16] where the detection of the central axis of the roads is done at low resolution while the detection of edges is done at high resolution. In [20], road extraction is done using quadratic active contours that incorporate strong geometrical constraints via the definition of interactions between contour points. Because they incorporate more specific information, these higher-order active contours are more robust to noise than conventional active contours, and permit a generic initialization that renders them automatic, contrary to classical contour approach. Results are promising but the method is time consuming.

In this report, we present a fully-automatic technique for line network extraction which is not based on a combination of several stages of image processing. We model the line network in the image by a spatial process in a compact $F \subset \mathbb{R}^2$, that is a random set of objects whose number of objects located in F is a random finite variable. Such models, introduced in image processing in [1], provide the same type of stochastic properties as those of Markov random fields, while incorporating strong geometric constraints. In [15, 14, 24], line network extraction is performed using spatial processes whose objects are interacting line segments described by three random variables: their midpoint, their length, and their orientation. The high-performance of this modeling has been verified on numerous examples. In particular, the "Quality Candy" prior model is especially suited to the extraction of road networks, leading to continuous line networks with a small curvature and few omissions and overdetections. Nevertheless, the detection of sinuous branches (characteristic of some rivers) is not enough accurate and the detection of "Y" intersections (*i.e.* intersections with an acute angle) remains difficult. We extend this modeling to more complex objects, such as in [21] for cell recognition, where objects are variable resolution deformable templates. More exactly, our new model, called CARtographic Oriented LIne Network Extraction (CAROLINE), is a spatial process where objects are interacting polylines composed by an unknown number of segments. The connection between segments is thus embedded into the object definition, whereas in [15, 24] the connection has to be defined up to a small constant (as, the probability of exact connection is null under the reference measure). Finally, the road and river junctions can be modeled easily through the definition of a pairwise interaction. At last, polylines can better fit sinuous line network than models based on segments through a relevant construction of the data term.

The CAROLINE model is presented in Section 2. The optimization - described in Section 3 - is done via a simulated annealing using a RJMCMC algorithm dedicated to the simulation of our spatial process. This algorithm is tested in Section 4 on remotely sensed images (aerial and satellite data).

2 CAROLINE model

2.1 Prior model

2.1.1 General framework

Point processes provide a rigorous framework based on measure theory to describe a scene by an unordered set of points in a compact $F \subset \mathbb{R}^d$ [27]. For $n \in \mathbb{N}$, let Ω_n be the set of configurations $\{x_1, \dots, x_n\}$ that consist of n unordered points of F. A **point process** on F is a mapping $\mathbf{X}$ from a probability space to the set of configurations $\Omega = \bigcup_{n=0}^{\infty} \Omega_n$, such that for all bounded Borel set $A \subseteq F$, the number of points $N_{\mathbf{X}}(A)$ falling in A is a finite random variable.

The canonical "completely random" point process is the **uniform Poisson point process**. Under the law of a Poisson point process of intensity λ, the number of points $N_{\mathbf{X}}(A)$ follows a Poisson law with mean $\lambda|A|$, and, given $N_{\mathbf{X}}(A) = n$, the n points are independently and uniformly distributed in A. The law of a Poisson process of intensity λ on $F \subset \mathbb{R}^d$ is defined by the following probability measure on $(\Omega, \mathcal{B})$[1]:

$$\mu(B) = \sum_{N=0}^{\infty} \frac{\lambda^N \, e^{-\lambda|F|}}{N!} \int_{F^N} \mathbf{1}_B(\{x_1, \dots, x_N\}) \, dx_1 \dots dx_N \tag{1}$$

where $B \in \mathcal{B}$.

To model the observed scene by a set of objects, we can augment a point process by adding extra information (*i.e.* object parameters) to each points. Such a process is called a **marked point process** or an **object process**. A marked point process on F, with marks in a space M, is a point process on $F \times M$ such that $N(A \times M) < \infty$ almost surely for any Borel bounded set $A \subset F$. In this context, the uniform Poisson process is a marked point process where points are distributed according to a uniform Poisson point process of intensity λ, and marks associated to each point are uniformly distributed in M. The law of this process on $F \subset \mathbb{R}^d$ is defined by the following probability measure:

$$\mu(B) = \sum_{N=0}^{\infty} \frac{e^{-\lambda F}}{N!} \int_{(F \times M)^N} \mathbf{1}_B(\{c_1, \dots, c_N\}) \, dx_1 dP_M(m_1) \dots dx_n dP_M(m_N) \tag{2}$$

where the $c_i = (x_i, m_i)$ are marked points of $F \times M$, and P_M is the uniform probability measure on the mark space M.

Although in most applications it is not realistic to assume that points are scattered randomly, Poisson processes are useful to build more complex models. Indeed, interactions can be introduced by specifying a **density** with respect to the reference measure μ. Let h be a nonnegative function on Ω. Then, the measure ν having a density h with respect to μ is defined by:

$$\nu(B) = \int_B h(C) \, \mu(dC) \tag{3}$$

If $0 < \nu(B) < \infty$, then ν can be normalized to provide a probability measure π defined by $\nu(B)/\nu(\Omega)$. In the context of object extraction from images, the density is usually factorized into two terms. First, geometrical and topological constraints are incorporated through a prior density h_p. Second, a data term h_d is used to fit the data. The complete density of the spatial process is then defined as follows:

$$h(C) \; \propto \; h_p(C) \; h_d(C) \tag{4}$$

where $C = \{c_1, \dots, c_n\}$ is a configuration of objects.

[1] $\mathcal{B}$ is the smallest σ-algebra such that for all Borel sets $A \subseteq F$ the mapping $\{x_1, \dots, x_n\} \mapsto N_{\mathbf{X}}(A)$ is measurable.

2.1.2 Model for an unknown number of polylines

The CAROLINE process is a random configuration of polylines, located in a compact set F of $\mathbb{R}^2$ corresponding to the observed scene, whose number N of polylines is unknown. In other words, the CAROLINE process is an object process on F of $\mathbb{R}^2$ where the objects are polylines. Each polyline c is described by:

- its initial point $p^1 \in F$;
- its width $e \in [e_{min}, e_{max}]$;
- an unknown number $n \in \{1, \ldots, n_{max}\}$ of segments;
- the segment lengths $l_j \in [L_{min}, L_{max}]$, $j = 1, \ldots, n$;
- the segment directions $\alpha_j \in]-\pi, \pi]$, $j = 1, \ldots, n$.

An example for $n = 3$ segments is given in Fig. 1.

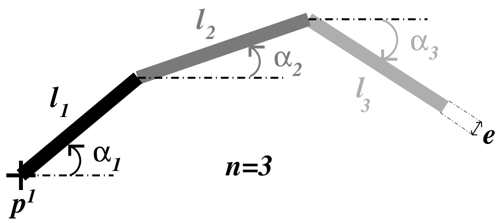


Figure 1: A typical object of the CAROLINE model.

We first define a reference process which is a uniform Poisson marked point process. Under the law of this process, the number N of polylines follows a Poisson law and, the initial points (points of the process) and polyline parameters (marks) are independently and uniformly distributed in their respective state space. The measure of this process is given by equation (2) where the uniform probability measure on the mark space is given by:

$$P_M(B) = \sum_{n=1}^{n_{max}} \frac{1}{n_{max}} \int_{[e_{min}, e_{max}]} \int_{V^n} \mathbf{1}_B(e, v_1, \ldots, v_n) \, \frac{d^n(v_1, \ldots, v_n)\, de}{|V|^n \, (e_{max} - e_{min})} \tag{5}$$

where B is an element of the tribu associated to the mark space $M = [e_{min}, e_{max}] \times \bigcup_{n=1}^{n_{max}} V^n$, where $V = [L_{min}, L_{max}] \times]-\pi, \pi]$ and $V^n = V \times \ldots \times V$ (n times).

To introduce an *a priori* on polyline shapes and interactions between polylines, we then specify the CAROLINE prior process by a prior density h_p with respect to the reference process law. The expression of h_p is given in Section 2.1.4 after a presentation of the possible interactions between polylines made in Section 2.1.3.

2.1.3 Polyline interactions

We consider two types of interaction.

The first one is based on a relation of **proximity** $\sim_p$ between polylines. This relation is defined as follows:

$$u \sim_p v \quad \Leftrightarrow \quad \begin{array}{l} \left(\exists j \in \{1, \ldots, n_u\} : d(p_u^j, v) < d_{max} \text{ and } d(p_u^{j+1}, v) < d_{max})\right) \\ \text{or } \left(\exists j \in \{1, \ldots, n_v\} : d(p_v^j, u) < d_{max} \text{ and } d(p_v^{j+1}, u) < d_{max})\right) \end{array} \tag{6}$$

where d corresponds to the Euclidean distance, n_u denotes the number of segments composing the polyline u, and p_u^j denotes the checking (or control) point number j describing u. In other words, two polylines are said close if two consecutive points of the first polyline are at a distance lower than d_{max} of the second polyline. An example of two polylines holding this relation is given in Figure 2. **This interaction is forbidden** in order to avoid overlapping of polylines. When this interaction occurs for a given polyline configuration C, the density $h_p(C)$ is thus equal to zero.

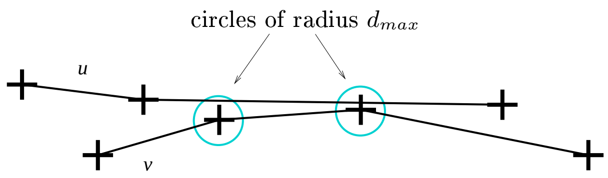


Figure 2: Proximity relation. $u \sim_p v$ because two consecutive points of v are at a distance lower than d_{max} of u.

The second interaction corresponds to the **connection** of a polyline via one of its extremities to another polyline. It is based on the Euclidean distance between the endpoints p_c^1 and p_c^{n+1} of a polyline c composed of n segments and another polyline or one edge of the compact F. If the distance $d(p_c^k, o)$ between p_c^k and o (polyline or edge) is lower than a threshold ϵ, c is said connected to o through p_c^k. Let $V_{C,F}(p_c^k)$ be the set of polylines and edges of C and F such that $d(p_c^k, o) < \epsilon$. We define three states of a polyline c according to the cardinality of the two sets $V_{C,F}(p_c^1)$ and $V_{C,F}(p_c^{n+1})$. A polyline c is said:

- **free**, if c is not connected by any of its extremities, *i.e.*:

$$V_{C,F}(c) = V_{C,F}(p_c^1) \cup V_{C,F}(p_c^{n+1}) = \emptyset$$

- **single**, if c is connected by only one of its extremities, *i.e.*:

$$V_{C,F}(c) \neq \emptyset \;, \quad \exists k : V_{C,F}(p_c^k) = \emptyset$$

- **double**, if c is connected by both of its extremities, *i.e.*:

$$V_{C,F}(p_c^1) \neq \emptyset \;, \quad V_{C,F}(p_c^{n+1}) \neq \emptyset$$

These three states are illustrated in Fig. 3.

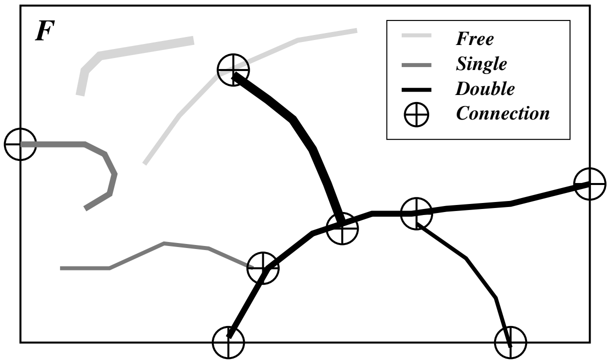


Figure 3: Polyline states with respect to connections.

2.1.4 CAROLINE prior density

The prior density h_p of a polyline configuration $C = \{c_1, \ldots, c_N\}$ can be written in a Gibbsian form as follows:

$$h_p(C) = \begin{cases} 0 \text{ , if } \exists\, c_i \in C, c_j \in C / c_i \sim_p c_j \\ \dfrac{1}{Z} \exp - \displaystyle\sum_{i=1}^{N} \left[U_1(c_i) + U_2(c_i, V_{C,F}(c_i)) \right] \text{ , if not} \end{cases} \tag{7}$$

where Z is an unknown normalizing constant, U_1 is a potential based on the object shape, and U_2 is a potential based on the polyline connections.

The energy term U_1 associated to a polyline c composed of n segments is written as follows:

$$U_1(c) = \begin{cases} +\infty \text{ , if } c \text{ self-intersects} \\ U_{11}(n) + \sum_{j=1}^{n} U_{12}(l_j) + \sum_{j=1}^{n-1} U_{13}(\alpha_j, \alpha_{j+1}) \text{ , if not} \end{cases} \tag{8}$$

with:

$$\begin{aligned} U_{11}(n) &= \frac{M_n}{(n+1)^2} \\ U_{12}(l_j) &= M_l \, \frac{L_{max} - l_j}{L_{max} - L_{min}} \\ U_{13}(\alpha_j, \alpha_{j+1}) &= M_\alpha \, (0.5 - \cos(\alpha_{j+1} - \alpha_j)) \end{aligned}$$

where M_n, M_l and M_α are positive weights. U_{11} **penalizes small** n, U_{12} **favors long segments**, and U_{13} **favors small curvature**. Moreover, **self-intersection is forbidden** by introducing a hard-core potential (*i.e.* infinite potential).

The energy term U_2 is written as follows:

$$U_2(c, V_{C,F}(c)) = U_{21}(c|V_{C,F}(c)) + \sum_{o \in V_{C,F}} U_{22}(c, o) \tag{9}$$

where U_{21} is based on the states of polylines and U_{22} is based on a measure of the quality of polyline connections.

U_{21} **penalizes free and single segments** through constant and positive potentials ω_f et ω_s:

$$U_{21}(c|V_{C,F}(c)\}) = \begin{cases} \omega_f \text{ , if } V_{C,F}(c) = \emptyset \\ \omega_s \text{ , if } V_{C,F}(c) \neq \emptyset \text{ , } \exists k : V_{C,F}(p_c^k) = \emptyset \\ 0 \text{ , if } V_{C,F}(p_c^1) \neq \emptyset \text{ , } V_{C,F}(p_c^{n+1}) \neq \emptyset \end{cases} \tag{10}$$

$U_{22}(c, o)$ **favors** the **connection** between an endpoint p_c^k and a polyline or an edge o through a negative potential, which is based on a function measuring the quality of the connection. This function depends on the distance between p_c^k and o. The more the distance decreases, the more the quality value increases. Let c a polyline connected to o through p_c^k. The quality of the connection is given by:

$$\sigma(< c, o >_{p_c^k}) = \frac{1}{\epsilon^2} \left(\frac{1 + \epsilon^2}{1 + d^2(p_c^k, o)} - 1 \right) \tag{11}$$

where ϵ is the connection threshold. The potential function U_{22} is then given by:

$$U_{22}(c|V_{C,F}(c)) = - \sum_{\substack{k = 1, n+1 \\ o : d(p_c^k, o) < \epsilon}} \sigma(< c, o >_{p_c^k}) \tag{12}$$

The **small distances of connection** are thus **favored** by a negative potential.

To sum up, the CAROLINE prior density favors long polylines with a small curvature that are connected with the rest of the network and small distances of connection while forbidding overlapping. This density specifies a **well-defined** marked point process, as the Ruelle's stability condition [22] is verified. Indeed, the stronger condition of **local stability** can easily be verified thanks to the introduction of a hard-core (*i.e.* infinite) potential with respect to the relation of proximity. To prove the local stability, a superior bound for the ratio $\frac{h_p(C \cup c)}{h_p(C)}$, for all C in Ω and c in $F \times M$ has to be found whenever $h_p(S) > 0$. The use of the hard-core potential induces a maximal paving of the compact F by a set of polylines: if a polyline c is added to such a set, then the proximity will be verified and the density will be null. There exists thus a threshold B such as the addition of a polyline can not induce more than B connections. The decrease of U_2 is therefore bounded. U_1 being bounded by below, there is a superior bound for $\frac{h_p(C \cup c)}{h_p(C)}$.

2.2 Data term

We build a data term based on the following assumptions:

- H_1: The grey level variation between the network and the background is large;
- H_2: The local average of the grey level inside the network is homogeneous.

To verify that a polyline is well-fitted to the data, we consider a mask of pixels composed of the set of pixels V corresponding to the polyline in the image and two collinear regions R_1 and R_2, positioned at a distance d from V, corresponding to the nearby background. Each pixel mask is divided into sections of a fixed number of pixels as shown in Fig. 4.

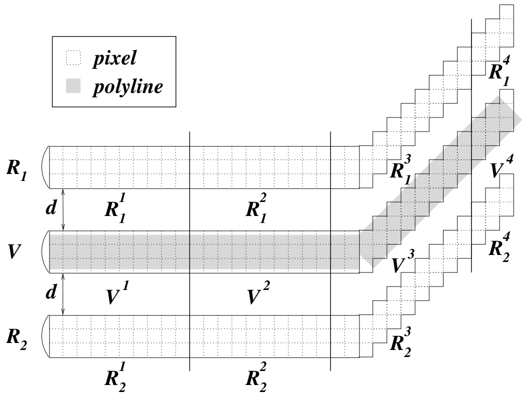


Figure 4: Pixel mask associated to a polyline.

Considering pixel values of each strip as a sample of a population, a Student t-test is used to determine if the averages of the two samples are significantly different. The formula for the t-test is a ratio. The top part of the ratio is the difference between the two sample means. The bottom part is a measure of the variability of the sample. Here is the t-test expression for two samples x and y :

$$\text{t-test}(x, y) = \frac{|\overline{x} - \overline{y}|}{\sqrt{\frac{\sigma_x^2}{n_x} + \frac{{\sigma^2}_y}{n_y}}} \tag{13}$$

where $\overline{x}$, σ_x and n_x (resp. $\overline{y}$, σ_y and n_y) refer to the sample mean, the sample standard deviation, and the number of observations of x (resp. y).

Firstly, the contrast hypothesis is checked for each section $M^i = \{V^i, R_1^i, R_2^i\}$. The test value t_c^i associated to M^i is the minimum of the two t-test values between the internal section V^i and the two external sections R_1^i and R_2^i:

$$t_c^i = \min_{l \in \{1,2\}} \left[\text{t-test}(R_l^i, V^i) \right] \tag{14}$$

Then, we perform a thresholding of t_c^i between τ_1 and τ_2 followed by a linear transformation from $[\tau_1, \tau_2]$ to $[-1, 1]$ in order to obtain a potential, $U_c(i)$, based on the measure of adequacy of H_1 for the section M^i:

$$U_c(i) = \begin{cases} 1 & \text{if } t_c^i < \tau_1 \\ 1 - 2\dfrac{t_c^i - \tau_1}{\tau_2 - \tau_1} & \text{if } \tau_1 \leq t_c^i \leq \tau_2 \\ -1 & \text{if } t_c^i > \tau_2 \end{cases} \tag{15}$$

Secondly, the homogeneity hypothesis H_2 is checked by computing the Student t-test values t_h^i between successive internal sections V^i and V^{i+1}:

$$t_h^i = \text{t-test}(V^i, V^{i+1}) \tag{16}$$

Then, we perform a thresholding of t_h^i between 1 and τ_h followed by a linear transformation from $[1, \tau_h]$ to $[-1, 1]$ in order to obtain a potential, $U_h(i, i+1)$, based on the measure of adequacy of H_2 for $\{V^i, V^{i+1}\}$:

$$U_h(i, i+1) = \begin{cases} -1 & \text{if } t_h^i < 1 \\ 1 - 2\dfrac{\tau_h - t_h^i}{\tau_h - 1} & \text{if } 1 \leq t_h^i \leq \tau_h \\ 1 & \text{if } t_h^i > \tau_h \end{cases} \tag{17}$$

Finally, the data potential associated to a polyline c is the following:

$$U_d(c) = p_c \sum_{i=1}^{I} U_c(i) + p_h \sum_{i=1}^{I-1} U_h(i, i+1) \tag{18}$$

where I is the number of sections composing the pixel mask associated to c, p_c and p_h are positive weights respectively associated to the contrast potential U_c and the homogeneity potential U_h. The total energy for a given configuration C is the sum of data potentials of each polyline belonging to C. The data term is thus given by:

$$h_d(C) \propto \exp\left(-\sum_{c \in C} U_d(c)\right) \tag{19}$$

where U_d is given by equation (18).

3 Optimization

3.1 Simulated annealing

To extract the line network from an image, we aim to find a configuration of polylines which maximizes the unnormalized process density h given by:

$$h(C) \propto h_p(C)\, h_d(C) \tag{20}$$

where h_p and h_d are respectively given by equation (7) and equation (19). This is a non convex problem for which a direct optimization is not possible given the large size of the state space that is $\cup_{N=0}^{\infty} \Omega_N$ where Ω_N is the set of configurations of N polylines. We propose to estimate this maximum by a simulated annealing scheme, which consists of successive simulations of the process distribution π_T specified by the density $h^{1/T}$, with T gradually dropping to 0. A proof of convergence is given in [26] when the decrease of temperature T is logarithmic. In practice, temperature decreases geometrically in order to reduce the computation time.

3.2 Simulation of spatial point processes

The algorithm chosen to simulate the unnormalized measure π_T is a Reversible Jump Markov Chain Monte Carlo (RJMCMC) algorithm [10, 11]. It consists of simulating a discrete Markov Chain of invariant measure π_T which performs small jumps between the spaces Ω_i. This iterative algorithm does not depend on the initial state. At each step, a transition from the current state S to a new state S' is proposed according to a proposition kernel $Q(S \to .)$. The transition is accepted with a probability $\alpha(S, S')$ given by the Green ratio. This acceptance ratio is computed so that the detailed balance condition is verified, condition under which this algorithm converges to π_T. This strong condition is given by:

$$\int_A \int_B \pi_T(dC)\, P(C, dC') = \int_B \int_A \pi_T(dC')\, P(C', dC) \tag{21}$$

where A and B are two sets of the tribu associated to Ω, and P is the transition kernel of the Markov chain C_t. Supposing that $\pi_T(.)\, Q(C \to .)$ has a finite density, $\mathcal{D}$, with respect to a symmetrical measure ψ on $\Omega \times \Omega$, the condition (21) is satisfied if:

$$\alpha(C, C')\, \mathcal{D}(C, C') = \alpha(C', C)\, \mathcal{D}(C', C) \tag{22}$$

As shown in [19] for the finite state space case, it is optimal to make the probability α as large as possible to reduce the autocorrelation of the Markov chain. Thus, we take:

$$\alpha(C, C') = \min\{1, R(C, C')\} \tag{23}$$

where R is the Green ratio given by:

$$R(C, C') = \frac{\mathcal{D}(C', C)}{\mathcal{D}(C, C')} \tag{24}$$

Thus, equality (22) is verified.

One interesting point of the Metropolis-Hastings-Green algorithm is that the proposal kernel Q can be decomposed into several sub-kernels q_i, each corresponding to a reversible move, as it has been proposed in [11]. Although it is sufficient to define uniform birth-and-death [10] in order to simulate spatial processes, it is important to define relevant moves in order to speed up the convergence of the Markov chain. In addition to a uniform birth-and-death, we have thus defined a birth-and-death of polylines containing only one segment based on data (off-line computation of the data term for small pixel masks) in order to propose segments correctly positioned. Proposing small perturbations is very useful when a polyline is already well positioned: dilation of a polyline; move of a point of a polyline (an endpoint or a point between two segments); add-and-remove of a segment at the end or at the beginning of a polyline; split-and-merge of segments. Moreover, we use a split-and-merge of polylines which is very relevant when two or more polylines are positioned on the same branch of the real line network. All these moves are illustrated in Fig. 5 and described in the following section.

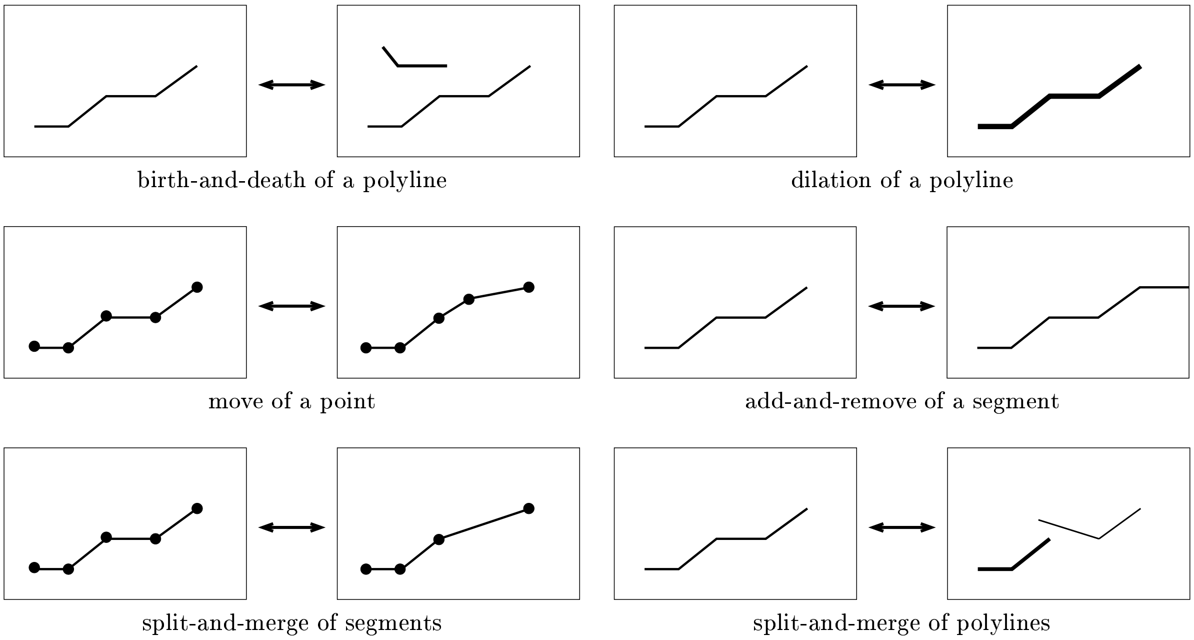


Figure 5: Reversible moves used in the proposed sampling algorithm.

3.3 Proposal kernels and Green ratio computation

In this section, proposal kernels dedicated to the simulation of the CAROLINE process are described. For each kernel, the explicit formula of the associated Green ratio is given.

3.3.1 Birth-and-death of a polyline

This section presents three types of birth-and-death of a polyline : a Uniform Birth-and-Death (UBD); a Birth-and-Death of a polyline Reduced to one single segment (BDR); and a BDR based on Data (BDRD).

UBD is the simplest proposal kernel which allows to make small jumps between spaces of different sizes. It consists of a uniform birth of a polyline in $F \times M$ - proposed with a probability p_b - and an uniform death (inverse proposal) in the set of segments C. The uniform birth consists of:

- uniformly drawing an initial point in F;
- uniformly drawing a width e in $[e_{min}, e_{max}]$;

- uniformly choosing a segment number n between 1 and n_{max};
- uniformly drawing the parameters describing each segment (length and direction) in V^n.

In the case of the birth of a polyline c, the Green ratio is given by:

$$R_{\mathrm{UBD}}(C, C \cup c) = \frac{p_d}{p_b} \frac{\lambda |F|}{N(C)+1} \frac{h(C \cup c)}{h(C)} \tag{25}$$

where p_b (resp. $p_d = 1 - p_b$) is the probability of choosing a birth (resp. a death), h is the process density, and λ is the intensity of the reference Poisson process. In the case of the death of a polyline c, the Green ratio is given by:

$$R_{\mathrm{UBD}}(C, C \setminus c) = \frac{p_b}{p_d} \frac{N(C)}{\lambda |F|} \frac{h(C \setminus c)}{h(C)} \tag{26}$$

The second type of birth corresponds to the proposition of a polyline composed of a single segment. The corresponding (BDR) reversible move corresponds thus to the proposition of such a birth with the probability $p_{b1\mathrm{s}}$ and the proposition of a death of a polyline composed of a single segment with the probability $p_{d1\mathrm{s}} = 1 - p_{b1\mathrm{s}}$. In the case of a sampling without data information, the birth of the polyline reduced to a segment is uniform: the proposed segment is uniformly drawn in $Z = F \times [e_{min}, e_{max}] \times [L_{min}, L_{max}] \times] - \pi, \pi]$. The death is also uniform: a segment is uniformly chosen in the set $E_1(C)$ of polylines in C reduced to one segment. In the case of a birth, the Green ratio is given by:

$$R_{\mathrm{BDR}}(C, C \cup c) = \frac{p_{d1\mathrm{s}}}{p_{b1\mathrm{s}}} \frac{\lambda |F|}{n_{max}\ (\sharp(E_1(C)) + 1)} \frac{h(C \cup c)}{h(C)} \tag{27}$$

where $\sharp(E_1(C))$ denotes the number of polylines in C reduced to one segment. Likewise, in the case of a death:

$$R_{\mathrm{BDR}}(C, C \setminus c) = \frac{p_{b1\mathrm{s}}}{p_{d1\mathrm{s}}} \frac{n_{max}\, \sharp(E_1(C))}{\lambda |F|} \frac{h(C \setminus c)}{h(C)} \tag{28}$$

Rather than uniformly proposing a new segment, it would be more relevant to use data information to propose more often segments that are well-positioned. So, we propose to replace the BDR kernel by a BDR based on Data (BDRD). The first step consists of computing the probabilities of linear structure presence in each pixel of the original image for a given number of orientations N_θ. For that, we store for every pixel $p_{i,\ i=1,\ldots,N_{pix}}$ and every orientation $\tilde{\theta}_{k,\ k=1,\ldots,N_\theta}$, the contrast potential $u_i^k \in [-1, 1]$ - given by equation (15) - corresponding to a segment of orientation $\tilde{\theta}_k$ centered on p_i. We then obtain, for every orientation θ_k, a map B_k defining an inhomogeneous (*i.e.* non uniform) birth kernel:

$$B_k(p_i) = \frac{2 - u_i^k}{\sum_{j=1}^{P} (2 - u_j^k)} \tag{29}$$

The weaker is the potential associated to a pixel p_i, the stronger is the probability $B_k(p_i)$ of proposing a segment of orientation $\tilde{\theta}_k$ centered on p_i.

The procedure for proposing of a new polyline c according to $B_1, \ldots, B_{N_\theta}$ is then the following:

- the width is uniformly drawn in $[e_{min}, e_{max}]$;
- the length l and the direction α of the first segment are uniformly drawn in $[L_{min}, L_{max}] \times] - \pi, \pi]$;
- a pixel p_i is drawn according to the map B_{k_α}, corresponding to the orientation $\tilde{\theta}_{k_\alpha}$ such that: $k_\alpha = \arg\min_j [|\alpha\, [\pi] - \tilde{\theta}_j|]$, where $[\pi]$ means modulo π;
- the segment center p is uniformly drawn in the square of $F \subset \mathbb{R}^2$ corresponding to the pixel p_i; the initial point p_c^1 of c is then computed from p, α, and l.

The inverse move consists of uniformly removing a polyline in $E_1(C)$. The Green ratios associated to the birth and the death of a polyline $c = (p_c^1, e, l, \alpha)$ are respectively given by:

$$R_{\text{BDRD}}(C, C \cup c) = \frac{p_{d1s}}{p_{b1s}} \frac{\lambda |F|}{N_{pix}\ B_{k_\alpha}(i_p)\ n_{max}\ (\sharp(E_1(C)) + 1)} \frac{h(C \cup c)}{h(C)} \tag{30}$$

$$R_{\text{BDRD}}(C, C \setminus c) = \frac{p_{b1s}}{p_{d1s}} \frac{N_{pix}\ B_{k_\alpha}(i_p)\ n_{max}\ \sharp(E_1(C))}{\lambda |F|} \frac{h(C \setminus c)}{h(C)} \tag{31}$$

where i_p is the pixel corresponding to projection of the center of (p_c^1, p_c^2) on the data image.

3.3.2 Simple Moves

The second kind of moves usually proposed to sample object processes is the modification of a randomly chosen object according to a symmetrical transformation. Let $\mathcal{T} = \{T_a : a \in E\}$ be a family of symmetrical transformations parameterized by a vector $a \in E$. If the modification of an object is done by applying T_a where a is uniformly chosen in E, the Green ratio is reduced to the density ratio:

$$R(C, C') = \frac{h(C')}{h(C)} \tag{32}$$

We propose here two symmetrical simple moves: the dilation of a polyline and the move of a checking point.

The dilation of a polyline consists of uniformly choosing $\delta_e \in [-\Delta_e, \Delta_e]$ and adding d_e to the polyline width e, with the additional condition that the new width is in $[e_{min}, e_{max}]$: $e' = e_{min} + ((e - e_{min} + \delta_e)\,[e_{max} - e_{min}])$, where $[.]$ denotes the modulo function.

The move of a point consists of uniformly choosing a checking point of the chosen polyline and proposing a translation of this checking point. This translation is a symmetrical transformation parameterized by a vector a uniformly drawn in a compact centered to the origin.

3.3.3 Add-and-remove of a segment

The Add-and-Remove (AR) perturbation consists of adding or removing a segment at the end or the beginning of a polyline. First, a polyline c is uniformly chosen in the current configuration. Second, the choice of the move type is done according to a probability based on the number of segments composing c. If the polyline is reduced to one segment, only the addition of a segment will be proposed. Likewise, if the polyline is composed of n_{max} segments, only the removal of a segment will be proposed. In the other cases, the removal and the addition of a segment are proposed with the probability 1/2. Once the move is chosen, an extremity is chosen with probability 1/2. In the addition case, a length l and a direction α are uniformly drawn in $V = [L_{min}, L_{max}] \times] - \pi, \pi]$. Let $c = (p, e, l_1, \alpha_1, \ldots, l_n, \alpha_n)$ be the chosen polyline. If the chosen extremity is the initial point $p = (x, y)$, then the new polyline is given by $c' = (p', e, l, \alpha, l_1, \alpha_1, \ldots, l_n, \alpha_n)$ where the new initial point is $p' = (x - l\cos(\alpha), y - l\sin(\alpha))$. If the chosen extremity is the final point, the new polyline is given by $c' = (p, e, l_1, \alpha_1, \ldots, l_n, \alpha_n, l, \alpha)$. In the removing case, if the chosen extremity is the initial point, then the new polyline is given by $c' = (p', e, l_2, \alpha_2, \ldots, l_n, \alpha_n)$ where $p' = (x + l_1\cos(\alpha_1), y + l_1\sin(\alpha_1))$ and, otherwise, $c' = (p, e, l_1, \alpha_1, \ldots, l_{n-1}, \alpha_{n-1})$.

In cases where the number of segments of the chosen polyline is between 2 and $n_{max} - 1$, the Green ratio is given by the ratio of the densities:

$$R_{\text{AR}}(C, C') = \frac{h(C')}{h(C)} \tag{33}$$

where C' is the configuration obtained by the perturbation of the current configuration C. In the other cases, the Green ratio is given by:

$$R_{\text{AR}}(C, C') = \frac{h(C')}{2\,h(C)} \tag{34}$$

The constant 1/2 intervenes as the addition of a segment to a polyline composed of one segment (resp. the removal of a segment of a polyline containing n_{max} segments) is done with probability 1 and the inverse move, *i.e.* the removal of a segment of a polyline composed of two segments (resp. the addition of a segment to a polyline composed of $n_{max}-1$ segments) is chosen with probability 1/2.

The combination of this sub-kernel with one of the two sub-kernels of birth-and-death of polylines reduced to one segment allows to guarantee the irreductibility of the Markov chain generated by the Monte Carlo algorithm. This combination allows to avoid the UBD kernel which, within an optimization framework, is not relevant: a birth of polyline composed of several segments is then mostly rejected.

3.3.4 Split-and-merge of segments

As for the translation, the addition and the removal of a checking point allow to perturb in a relevant way a polyline approximatively well-positioned. These moves define a reversible move called Split-and-Merge of Segments (SMS) which is illustrated by figure 6.

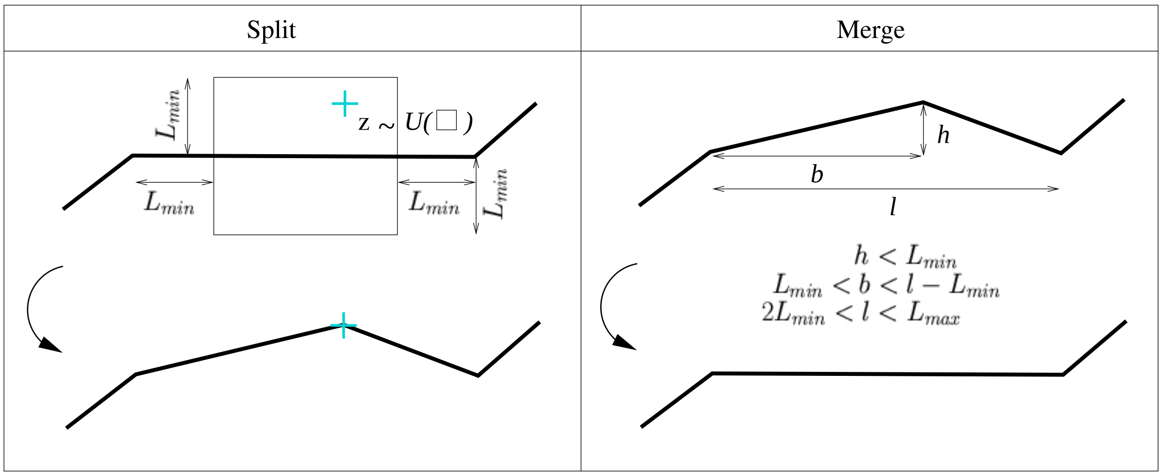


Figure 6: Split-and-merge of segments.

a) Split

Let $s=(p^j,p^{j+1})$ be a segment of a polyline c. The splitting of s is only proposed if the length of s, l, is larger than $2L_{min}$, and if c is not composed of n_{max} segments. We use an auxiliary variable Z in order to obtain two new segments $s'_1=(p^j,p')$ and $s'_2=(p',p^{j+1})$ such as p' is located in the rectangle of length $l-2L_{min}$ and width $2L_{min}$ and whose main axis corresponds to the segment s. In this way, the lengths of the new segments are in $[L_{min},L_{max}]$. The variable Z corresponds to a uniform drawing of points in the rectangle of length $l-2L_{min}$ and width $2L_{min}$:

$$Z = \begin{bmatrix} H \sim \mathcal{U}([-L_{min}, L_{min}]) \\ B \sim \mathcal{U}([L_{min}, l-L_{min}]) \end{bmatrix} \tag{35}$$

>From the generated vector $z=(h,b)$, we obtain the parameters $v_1=(l_1,\alpha_1)$ describing the segment s'_1 using the diffeomorphism defined by:

$$v_1 = \eta_v(h,b) = \begin{bmatrix} \sqrt{h^2+b^2} \\ \alpha + \arctan(\frac{h}{b}) \end{bmatrix} \tag{36}$$

where the parameters $v=(l,\alpha)$ of s are fixed. We obtain then the parameters $v_2=(l_2,\alpha_2)$ describing the segment s'_2 from v_1 et $v=(l,\alpha)$ as follows:

$$v_2 = T(v,v_1) = \begin{bmatrix} \sqrt{(l\sin(\alpha) - l_1\ \sin(\alpha_1))^2 + (l\ \cos(\alpha) - l_1\ \cos(\alpha_1))^2} \\ \arctan(\dfrac{l\sin(\alpha) - l_1\ \sin(\alpha_1)}{l\cos(\alpha) - l_1\ \cos(\alpha_1)}) \end{bmatrix} \tag{37}$$

b) Merge

Let $s_j=(p^j,p^{j+1})$ and $s_{j+1}=(p^{j+1},p^{j+2})$ the two consecutive segments of a polyline c. Merging these two segments consists of replacing s_j and s_{j+1} by $s'_j=(p^j,p^{j+2})$. The merging of two segments is only proposed if the point p^{j+1} is located in the rectangle associated to $s_{l,l+1}$ for a split proposition. The conditions to verify are thus the following:

- $d(p^j, p^{j+2}) \in]2L_{min}, L_{max}]$;
- the distance of p^{j+1} to the segment $s'_j = (p^j, p^{j+2})$ is lower than L_{min};
- the orthogonal projection of p^{j+1} on s'_j is located at a distance b of p^j that is between $2L_{min}$ and $l - L_{min}$.

c) Proposal kernel

Let C be the current configuration. There are $N_S(C)$ segments that can be split and $N_M(C)$ couple of segments that can be merged. There are thus $N_T(C) = N_S(C) + N_M(C)$ possible moves of type split-and-merge of segments. The first step consists of uniformly drawing a move among the $N_T(C)$ possible moves. If the chosen move is a split (resp. merge), we proceed as explained in a) (resp. b)). The proposal kernel Q_{SMS} is thud given by:

$$Q_{\text{SMS}}(C \to A) = \sum_{s_i \in S(C)} \frac{q_i^S(C, A)}{N_T(C)} + \sum_{(s_i, s_{i+1}) \in M(C)} \frac{q_i^M(C, A)}{N_T(C)} \tag{38}$$

where $S(C)$ denotes the set of segments that can be split, $M(C)$ denotes the set of consecutive segments that can be merged, $q_i^S(C, A)$ corresponds to the splitting of the segment s_i, and $q_i^M(C, A)$ corresponds the merging of the couple (s_i, s_{i+1}). The split part $q_i^S(C, A)$ is given by:

$$q_i^S(C, A) = \int_{\Sigma_i} \mathbf{1}_A(S_i(C, z)) \, \frac{dz}{2L_{min}\,(l_i - 2L_{min})} \tag{39}$$

where $\Sigma_i = [-L_{min}, L_{min}] \times [L_{min}, l_i - L_{min}]$ is the compact in which the variable Z is drawn and $S_i(C)$ is a configuration of polylines where the parameters $v_i = (l_i, \alpha_i)$ of the segment s_i are replaced by $(l_1, \alpha_1) = \eta_{v_i}(z)$ and $(l_2, \alpha_2) = T(v_i, \eta_{v_i}(z))$. The merge part $q_i^M(C, A)$ of the kernel is given by:

$$q_i^M(C, A) = \mathbf{1}_A(M_i(C)) \tag{40}$$

$M_i(C)$ is a configuration of polylines where the parameters of the couple (s_i, s_{i+1}) are replaced by the parameters of the segment obtained by the merging of s_i and s_{i+1}.

d) Green ratio computation

The symmetrical measure ψ on $\Omega \times \Omega$ chosen to derivate the measure πQ_{SMS} is the following:

$$\begin{aligned} \psi(A, B) \;=\; & \int_A \sum_{s_i \in S(C)} \int_{\Sigma_i} \mathbf{1}_B(S_i(C, z)) \, \frac{dz}{|V|} \, d\mu(C) \\ & + \int_A \sum_{(s_i, s_{i+1}) \in M(C)} \mathbf{1}_B(M_i(C)) \, |J_{\phi^{-1}}(v_i, v_{i+1})| \, d\mu(C) \end{aligned} \tag{41}$$

where $V = [L_{min}, L_{max}] \times [-\pi, \pi]$, and ϕ is the diffeomorphism corresponding to the following variable change:

$$(v_1, v_2) \xleftarrow{\phi} (z, v)$$

where (v_1, v_2) correspond to the parameters of the two segments obtained after splitting the segment of parameters $v = (l, \theta)$ using the auxiliary variable z. This diffeomorphism is given by:

$$\phi(z, v) = (\eta_v(z), T(v, \eta_v(z))) \tag{42}$$

where η_v and T are respectively given by equations (36) and (37). $J_{\phi^{-1}}$ denotes the Jacobian of ϕ^{-1} whose determinant is given by:

$$|J_{\phi^{-1}}(v_1, v_2)| = \frac{l_1\, l_2}{l} \tag{43}$$

Finally, the Green ratio corresponding to the split of a segment of length l in two segments of lengths l_1 and l_2 is given by:

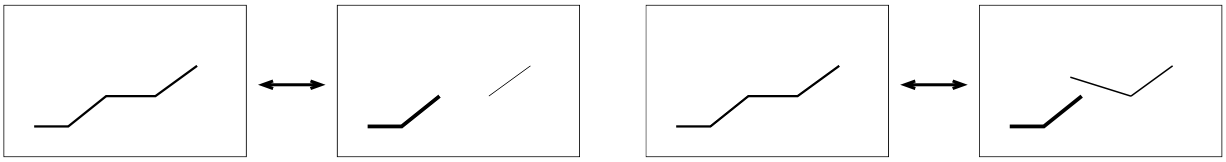
$$R_{\mathrm{SMS}}(C,C') = \frac{N_T(C)}{N_T(C')}\frac{L_{min}(l-2L_{min})}{\pi\,(L_{max}-L_{min})}\frac{l}{l_1\,l_2}\frac{h(C')}{h(C)} \tag{44}$$

Likewise, in the merging case, the Green ratio is given by:

$$R_{\mathrm{SMS}}(C,C') = \frac{N_T(C)}{N_T(C')}\frac{\pi\,(L_{max}-L_{min})}{L_{min}(l-2L_{min})}\frac{l_1\,l_2}{l}\frac{h(C')}{h(C)} \tag{45}$$

3.3.5 Split-and-merge of polylines

We propose two perturbations of type split-and-merge of polylines. These two reversible moves are illustrated in figure 7. The first perturbation (SMP1) consists of adding a segment linking two polylines and, reversely, removing a segment of a polyline. The second perturbation (SMP2) consists of removing a connection between two segments and, reversely, creating a connection by moving of an endpoint of a polyline.



(SMP1) (SMP2)

Figure 7: Perturbations of type split-and-merge of polylines.

a) SMP1 move

Let S_{j,δ_e} be the perturbation which splits a polyline by removing a segment j of a polyline of width e and composed by at least three segments and which allows to define two polylines whose widths are equal to $e+\delta_e$ and $e-\delta_e$:

$$S_{j,\delta_e}(p,e,v_1,\ldots,v_n) = \{(p^1, e+\delta_e, v_1,\ldots,v_{j-1}), (p^{j+1}, e-\delta_e, v_{j+1},\ldots,v_n)\} \tag{46}$$

The variation of width δ_e is uniformly drawn in a compact such as the two new widths are in $[e_{min}, e_{max}]$:

$$\delta_e \sim \mathcal{U}([-B(e),B(e)]) \text{ , where } B(e) = \min\{e-e_{min}, e_{max}-e\} \tag{47}$$

The inverse perturbation concerns the couple of polylines (c_i, c_j) whose final point $p_i^{n_i+1}$ of c_i and initial point p_j^1 of c_j verify:

$$L_{min} \le d(p_i^{n_i+1}, p_j^1) \le L_{max} \tag{48}$$

Moreover, the sum of the segment number (n_i+n_j) has to be lower than n_{max}:

$$n_i + n_j < n_{max} \tag{49}$$

The merge move $M(c_i,c_j)$ for $c_i = (p_i^1, e_i, (v_i^k)_{k=1..n_i})$ et $c_j = (p_j^1, e_j, (v_j^k)_{k=1..n_j})$ is defined as follows:

$$M(c_i,c_j) = (p_i^1, \frac{e_i+e_j}{2}, (v_i^k)_{k=1..n_i}, v_{ij}, (v_j^k)_{k=1..n_j}) \tag{50}$$

where v_{ij} corresponds of the parameters of the segment $(p_i^{n_i+1}, p_j^1)$.

This definition of the split-and-merge only includes the couple of polylines whose final point and initial point verify the proximity condition (48). It would be relevant to merge also couples whose initial extremities or final extremities verify this condition. A first solution would be to add a new proposition kernel in the RJMCMC algorithm that correspond to the inversion move, *i.e.* move that inverses a polyline as follows:

$$(p^1, e, v_1, \ldots, v_n) \leftarrow (p^{n+1}, e, v'_n, \ldots, v'_1) \tag{51}$$

where $v'_j = (l_j, \alpha_j - \pi)$ if $\alpha_j > 0$, and $v'_j = (l_j, \alpha_j + \pi)$ otherwise.

Here, we do not use such a kernel but we directly incorporate this move into the kernel Q_{SMP1}. Indeed, we add a step that consists of randomly proposing a permutation of the sense of the polyline(s). For a split move S_{j,δ_e}, the resulting pair of polylines is thus uniformly chosen among the following four pairs:

$$
\begin{array}{rcl}
S^1_{j,\delta_e}(p^1, e, v_1, \dots, v_n) & = & \{(p^1, e+\delta_e, v_1, \dots, v_{j-1}), (p^{j+1}, e-\delta_e, v_{j+1}, \dots, v_n)\} \\
S^2_{j,\delta_e}(p^1, e, v_1, \dots, v_n) & = & \{(p^j, e+\delta_e, v'_{j-1}, \dots, v'_1), (p^{j+1}, e-\delta_e, v_{j+1}, \dots, v_n)\} \\
S^3_{j,\delta_e}(p^1, e, v_1, \dots, v_n) & = & \{(p^j, e+\delta_e, v'_{j-1}, \dots, v'_1), (p^{n+1}, e-\delta_e, v'_n, \dots, v'_{j+1})\} \\
S^4_{j,\delta_e}(p^1, e, v_1, \dots, v_n) & = & \{(p^1, e+\delta_e, v_1, \dots, v_{j-1}), (p^{n+1}, e-\delta_e, v'_n, \dots, v'_{j+1})\}
\end{array}
\tag{52}
$$

We now consider that the merging between two polylines c_i and c_j can be done by four pairs of endpoints: $\{p_i, p_j\}_1 = \{p_i^{n_i+1}, p_j^1\}$; $\{p_i, p_j\}_2 = \{p_i^1, p_j^1\}$; $\{p_i, p_j\}_3 = \{p_i^1, p^1_{n_j+1}\}$; $\{p_i, p_j\}_m = \{p_i, p_j\}_4 = \{p_i^{n_i+1}, p^1_{n_j+1}\}$. Let $\{p_i, p_j\}_k$ be a pair of extremities that can be linked to merge c_i and c_j. The resulting polyline is uniformly chosen among the following two polylines:

$$
\begin{array}{l}
M^1_k(c_i, c_j) = \left\{
\begin{array}{llllll}
(p_i^1, & \frac{e_i+e_j}{2}, & ({v_i}^l)_{l=1..n_i}, & v_{ij}, & ({v_j}^l)_{l=1..n_j}) & \text{if } k=1, \ i.e. \ \{p_i, p_j\}_k = \{p_i^{n_i+1}, p_j^1\} \\
(p_i^{n_i+1}, & \frac{e_i+e_j}{2}, & ({v'_i}^l)_{l=n_i..1}, & v_{ij}, & ({v_j}^l)_{l=1..n_j}) & \text{if } k=2, \ i.e. \ \{p_i, p_j\}_k = \{p_i^1, p_j^1\} \\
(p_i^{n_i+1}, & \frac{e_i+e_j}{2}, & ({v'_i}^l)_{l=n_i..1}, & v_{ij}, & ({v'_j}^l)_{l=n_j..1}) & \text{if } k=3, \ i.e. \ \{p_i, p_j\}_k = \{p_i^1, p^1_{n_j+1}\} \\
(p_i^1, & \frac{e_i+e_j}{2}, & ({v_i}^l)_{l=1..n_i}, & v_{ij}, & ({v'_j}^l)_{l=n_j..1}) & \text{if } k=4, \ i.e. \ \{p_i, p_j\}_k = \{p_i^{n_i+1}, p^1_{n_j+1}\}
\end{array}
\right. \\
M^2_k(c_i, c_j) = \left\{
\begin{array}{llllll}
(p_i^{n_j+1}, & \frac{e_i+e_j}{2}, & ({v'_j}^l)_{l=n_j..1}, & v_{ji}, & ({v'_i}^l)_{l=n_i..1}) & \text{if } k=1, \ i.e. \ \{p_i, p_j\}_k = \{p_i^{n_i+1}, p_j^1\} \\
(p_j^{n_j+1}, & \frac{e_i+e_j}{2}, & ({v'_j}^l)_{l=n_j..1}, & v_{ji}, & ({v_i}^l)_{l=1..n_i}) & \text{if } k=2, \ i.e. \ \{p_i, p_j\}_k = \{p_i^1, p_j^1\} \\
(p_j^1, & \frac{e_i+e_j}{2}, & ({v_j}^l)_{l=1..n_j}, & v_{ji}, & ({v_i}^l)_{l=1..n_i}) & \text{if } k=3, \ i.e. \ \{p_i, p_j\}_k = \{p_i^1, p^1_{n_j+1}\} \\
(p_j^1, & \frac{e_i+e_j}{2}, & ({v_j}^l)_{l=1..n_j}, & v_{ji}, & ({v'_i}^l)_{l=n_i..1}) & \text{if } k=4, \ i.e. \ \{p_i, p_j\}_k = \{p_i^{n_i+1}, p^1_{n_j+1}\}
\end{array}
\right.
\end{array}
\tag{53}
$$

Let N_S be the number of segments whose removal allows the generation of two polylines and N_M the number of pairs of extremities whose linking allows the generation of a new polyline. A move is uniformly chosen among the $N_T = N_S + N_M$ possible moves. The kernel Q_{SMP1} is then given by:

$$
\begin{array}{rcl}
Q_{\text{SMP1}}(C \to A) & = & \displaystyle\sum_{s_i^j \in \mathcal{S}(C)} \frac{1}{N_T(C)} \sum_{k=1}^{4} \frac{1}{4} \int_{-B(e_i)}^{B(e_i)} \mathbf{1}_A((C \setminus c_i) \cup S^k_{j,\delta_e}(c_i)) \, \frac{d\delta_e}{2B(e_i)} \\
& & \displaystyle + \sum_{\{p_i, p_j\}_k \in \mathcal{M}(C)} \frac{1}{N_T(C)} \sum_{m=1}^{2} \frac{1}{2} \mathbf{1}_A((C \setminus \{c_i, c_j\}) \cup M^m_k(c_i, c_j))
\end{array}
\tag{54}
$$

where s_i^j denotes the segment j of the polyline c_i of width e_i, p_i^k denotes the point k of c_i, $\mathcal{S}(C)$ is the set of segments whose removal allows the generation of two polylines, and $\mathcal{M}(C)$ the set of segments that can generate a polyline by adding a segment linking them. The kernel is thus given by:

$$
\begin{array}{rcl}
Q_{\text{SMP1}}(C \to A) & = & \displaystyle \frac{1}{8N_T(C)} \sum_{c_i \in C} \frac{1}{B(e_i)} \sum_{j=2}^{n_i-1} \sum_{k=1}^{4} \int_{-B(e_i)}^{B(e_i)} \mathbf{1}_A((C \setminus c_i) \cup S^k_{j,\delta_e}(c_i)) \, d\delta_e \\
& & \displaystyle + \frac{1}{2N_T(C)} \sum_{\{c_i, c_j\} \in C} \sum_{k=1}^{4} \sum_{m=1}^{2} \mathbf{1}_A((C \setminus \{c_i, c_j\}) \cup M^m_k(c_i, c_j))
\end{array}
\tag{55}
$$

The measure ψ on $\Omega \times \Omega$ chosen to derivate the measure πQ_{SMP1} is a symmetrical measure concentrated on $\bigcup_{n=0}^{\infty} \{\{\Omega_N \times \Omega_{N+1}\} \cup \{\Omega_{N+1} \times \Omega_N\}\}$. Let A and B be sets of the tribu of Ω such as $A \subseteq E_N$ and $B \subseteq E_{N+1}$. The measure ϕ is then given by:

$$
\begin{array}{rcl}
\psi(A, B) & = & \displaystyle \int_A \sum_{i=1}^{N} \sum_{j=2}^{n_i-1} \sum_{k=1}^{4} \int_{-B(e_i)}^{B(e_i)} \mathbf{1}_B((C \setminus c_i) \cup S^k_{j,\delta_e}(c_i)) \, d\delta_e \, d\mu(C) \\
\psi(B, A) & = & \displaystyle \int_B \sum_{i=1}^{N} \sum_{\substack{j=1 \\ j \neq i}}^{N} \sum_{k=1}^{4} \frac{n_{max}(e_{max} - e_{min})}{2 d_k(c_i, c_j) \lambda |V|} \mathbf{1}_A((C \setminus \{c_i, c_j\}) \cup M^1_k(c_i, c_j) d\mu(C) \\
& = & \displaystyle \int_B \sum_{\{c_i, c_j\}} \sum_{m=1}^{2} \sum_{k=1}^{4} \frac{n_{max}(e_{max} - e_{min})}{2 d_k(c_i, c_j) \lambda |V|} \mathbf{1}_A(C \setminus \{c_i, c_j\}) \cup M^m_k(c_i, c_j) d\mu(C)
\end{array}
\tag{56}
$$

where Σ denotes the state space associated to the polyline width, and $d_k(c_i, c_j)$ denotes the distance between the two points of $\{p_1, p_2\}_k$. The factor $\frac{1}{2d_k(c_i,c_j)}$ is due to the variable change:

$$(e_1, e_2, p') \longleftarrow (e, \delta_e, l_j, \alpha_j)$$

where e_1 and e_2 are the width of c_1 and c_2 obtained by removing of the segment j of a polyline of width e, and p' corresponds to the generated initial point.

Finally, the Green ratio corresponding of the split of a polyline of width e by removing a segment of length l is given by:

$$R_{\text{SMP1}}(C, C') = \frac{N_T(C)}{N_T(C')} \, \frac{8\, l\, \lambda\, |V| \, \min(e - e_{min}, e_{max} - e)}{n_{max}(e_{max} - e_{min})} \, \frac{h(C')}{h(C)} \tag{57}$$

where $|V| = 2\pi(L_{max} - L_{min})$. In the reverse case, the Green ratio is given by:

$$R_{\text{SMP1}}(C, C') = \frac{N_T(C)}{N_T(C')} \, \frac{n_{max}(e_{max} - e_{min})}{8\, l\, \lambda\, |V| \min(e - e_{min}, e_{max} - e)} \, \frac{h(C')}{h(C)} \tag{58}$$

b) SMP2 move

The second SMP perturbation was built to merge close polylines that do not verify the proximity condition. Here, the merge is done by removing the first or the last segment of a polyline followed by adding a linking segment. The new width is equal to the mean of two widths of the merged polylines. For a couple of polylines (c_i, c_j) verifying $n_i + n_j \leq n_{max}$, we have height possibilities of new unordered polylines as shown in figure 8.

Among these height possibilities, we only keep the propositions of segments whose length is between L_{min} and L_{max}, which allows to generate well-defined polylines. For a given merge, a polyline is uniformly chosen among two polylines composed by the same segments but in a different order

The split concerns the polylines composed by at least two segments. A checking point p_i^j is uniformly chosen among the points $(p_i^2, \dots, p_i^{n_i})$. The polyline (of width e) is then cut at this point. The widths of the two resulting polylines are equal to $e + \delta_e$ and $e - \delta_e$, where δ_e is uniformly drawn in $[-B(e), B(e)]$ defined in equation (47). For one of the two polylines, chosen with probability 1/2, the checking point is regenerated according to a uniform drawing of a length l and a direction α. Four ordered polylines can then be generated by reversing of the segment order. There are thus height possible splits from a checking point p_i^j.

Let $N_M(C)$ be the number of points that allows the merging of two polylines of C and $N_S(C)$ the number of points that allows the splitting of a polyline of C. A move is uniformly chosen among the $N_T(C) = N_F(C) + N_D(C)$ possible moves. The kernel is then given by:

$$\begin{aligned} Q_{\text{SMP2}}(C \to A) \quad = \quad & \sum_{c_i \in C} \sum_{j=2}^{n_i} \frac{1}{N_T(C)} \sum_{k=1}^{8} \frac{1}{8} \int_{-B(e_i)}^{B(e_i)} \int_V \mathbf{1}_A((C \setminus c_i) \cup S_{j,\delta_e}^k(c_i)) \, \frac{dv \; d\delta_e}{|V| \, 2B(e_i)} \\ & + \sum_{\{c_i, c_j\} \in C} \sum_{k=1}^{8} \frac{1}{N_T(C)} \sum_{m=1}^{2} \frac{1}{2} \mathbf{1}_A((C \setminus \{c_i, c_j\}) \cup M_k^m(c_i, c_j)) \end{aligned} \tag{59}$$

The Green ratio corresponding to a split of a polyline of width e by replacing a segment of length l, is then given by:

$$R(C, C') = \frac{N_T(C)}{N_T(C')} \, \frac{16\, l\, \lambda\, |V| \min(e - e_{min}, e_{max} - e)}{n_{max}(e_{max} - e_{min})} \, \frac{h(C')}{h(C)} \tag{60}$$

Likewise, the Green ratio corresponding to a merge of two polylines replacing an initial or final segment by a new segment of length l, is given by:

$$R(C, C') = \frac{N_T(C)}{N_T(C')} \, \frac{n_{max}(e_{max} - e_{min})}{16\, l\, \lambda\, |V| \min(e - e_{min}, e_{max} - e)} \, \frac{h(C')}{h(C)} \tag{61}$$

3.4 Study on the behavior of the RJMCMC algorithm

In this section, we check the behavior of the MCMC algorithm for sampling a uniform Poisson process with $n_{max} = 10$ and $\lambda|F| = 100$. For that, we consider the empirical averages of N, the total number of polylines, and $\{N_i\}_{i=1..10}$, the number of polylines composed of i segments. The means are computed from a fixed number of iterations I_0 (here, $I_0 = 30000$). From I_0, we consider a sample every P iterations (here, $P = 10000$). This

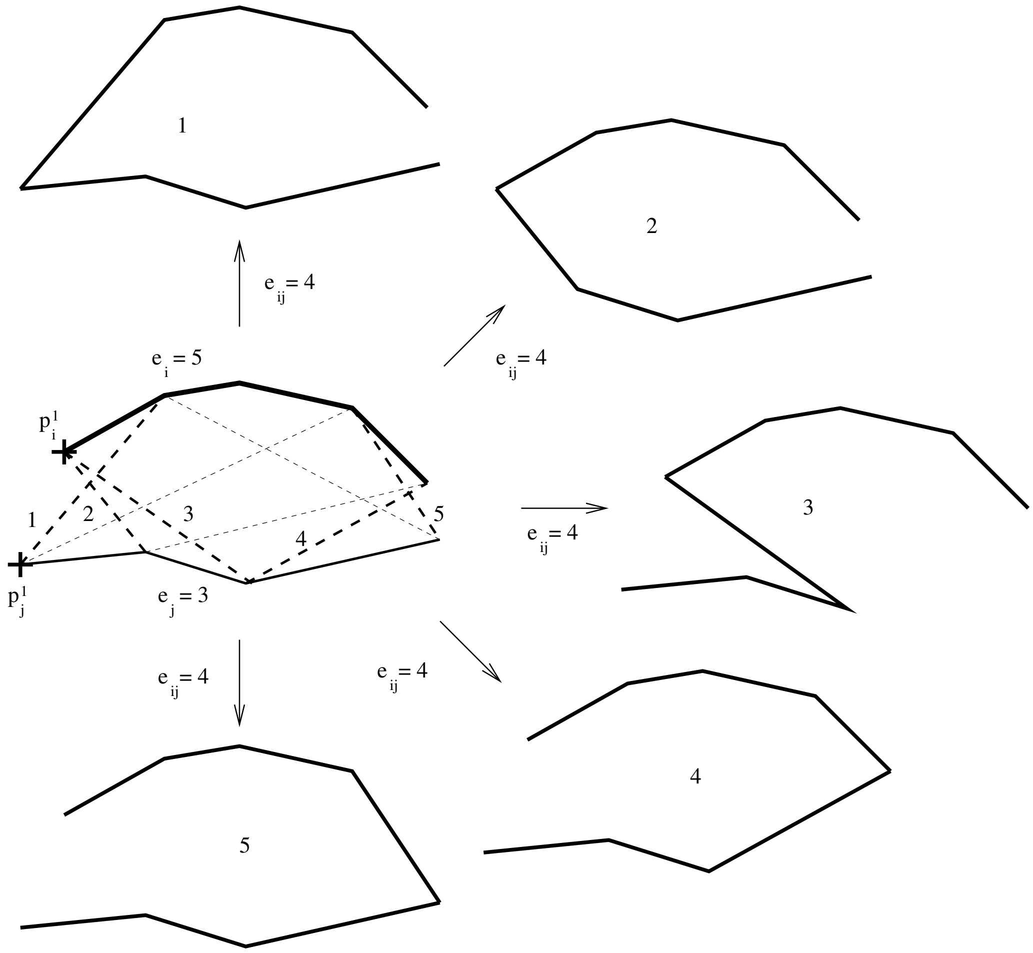


Figure 8: Possible merges of a pair of polylines. Only five merges are proposed because the other ones do not respect the condition on the length of the new segment.

sampling allows to reduce the effect of the strong correlation between consecutive samples. The algorithm is stopped when the difference of empirical values is sufficiently small during thirty successive steps.

In order to show that a UBD kernel can be replaced by a combination of a BDR kernel and an AR kernel, we first check the behavior of the MCMC algorithm using the kernel $Q = 1/2\, Q_{\mathrm{BDR}} + 1/2\, Q_{\mathrm{AR}}$, where Q_{BDR} is the BDR kernel without radiometrical information, and Q_{AR} is the AR kernel. The algorithm converged in less than 34 seconds with a processor 2 GHz (3.5×10^6 iterations). The empirical averages computed during the algorithm are shown in figure 9. The empirical error done on the point number is weak: about 0.2 for a theoretical value of $E(N) = \lambda|F| = 100$. Moreover, the empirical values of the expectation of the number of polyline of size i ($i = 1, \dots, n_{max}$) are close to the theoretical value obtained by the hypothesis of a uniform law on the number of segments composing a polyline:

$$P(i) = \frac{1}{10}\ \forall i = 1..10 \ \Rightarrow\ E[N_i] = \frac{E[N]}{10} = 10\ \forall i = 1..10$$

We have then tested each sub-kernel proposed in the section 3.3 by adding this sub-kernel to the kernel Q, and using the same procedure. The empirical errors are lower than 2% for each empirical mean.

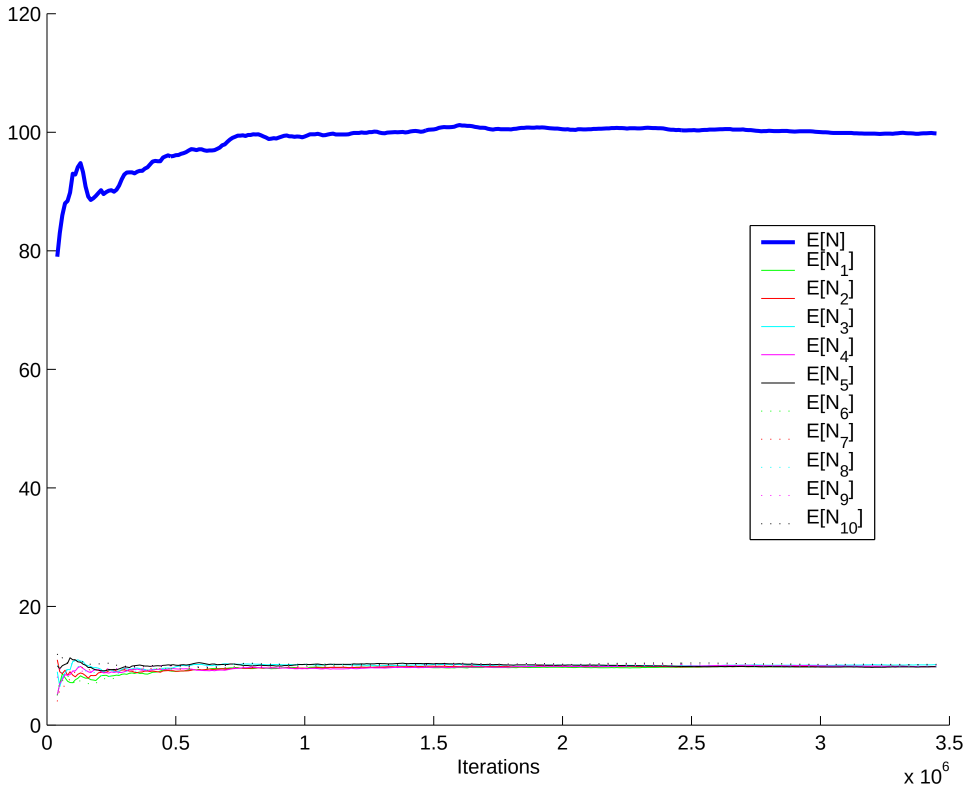


Figure 9: Empirical averages of N, the total number of polylines, and $\{N_i\}_{i=1..10}$, the number of polylines composed of i segments, with respect to the number of iterations.

4 Results

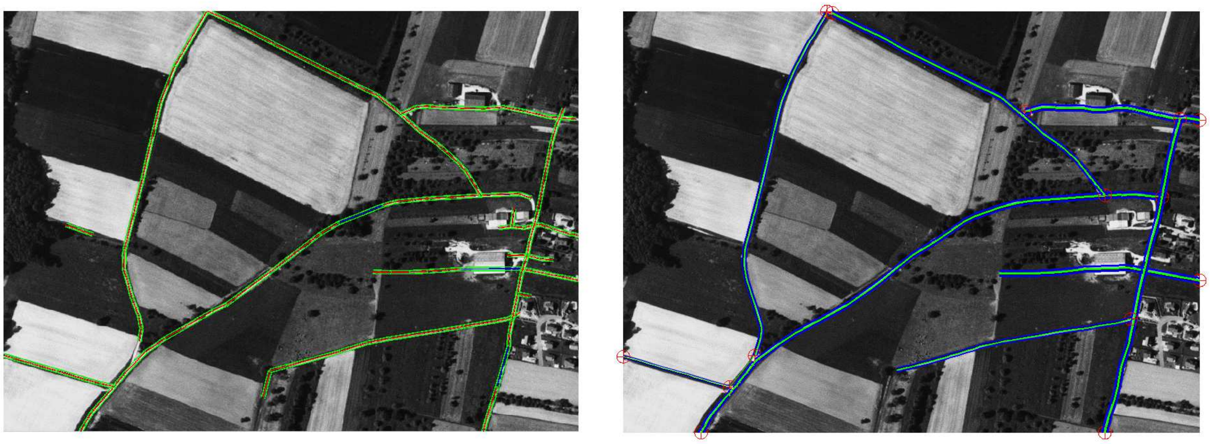
This section presents results of the simulated annealing described in Section 3 on four remotely sensed images, with an initialization by the empty configuration. For each image, the result obtained using the "Quality Candy" process - proposed in [15] - is also provided.
Even if the perturbations defined in the section 3.3 accelerate convergence and allow to avoid local minima, it is necessary to use a very slow decrease of the temperature - in comparison with the segment process "Quality Candy" - in order to obtain the wholeness of the network. This is why we use here an **adaptive decrease of the temperature**, which allows to decrease as much as possible the temperature while remaining close to the balance. During m iterations the temperature is constant and we compute the empirical average of the system energy. At the end of this period the energy average is compared to the one of the previous period. At the next period, the temperature decreases only if there is an increase of the energy.

Globally, both processes - "CAROLINE" and "Quality Candy"- provide a **continuous line networks with few omissions and few overdetections**. The presented results show these models are fitted to different type of data and objects to be extracted: satellite and aerial images, optical and radar images, river and road networks. Moreover, their robustness with respect to noise is high due to the introduction of strong geometrical constraints in the prior density. In particular, these models are robust to geometrical noise, that is to say noise characteristic to the observed scene. This is shown on the aerial image presented in figure 10 where trees interfere with the detection of road sections. As shown in figure 11, the prior models allow to detect all the sections occluded by the presence of trees.

Figure 10: Aerial image of size 1784×1304 pixels (50 centimeters of resolution) provided by the French Mapping Institute (IGN).

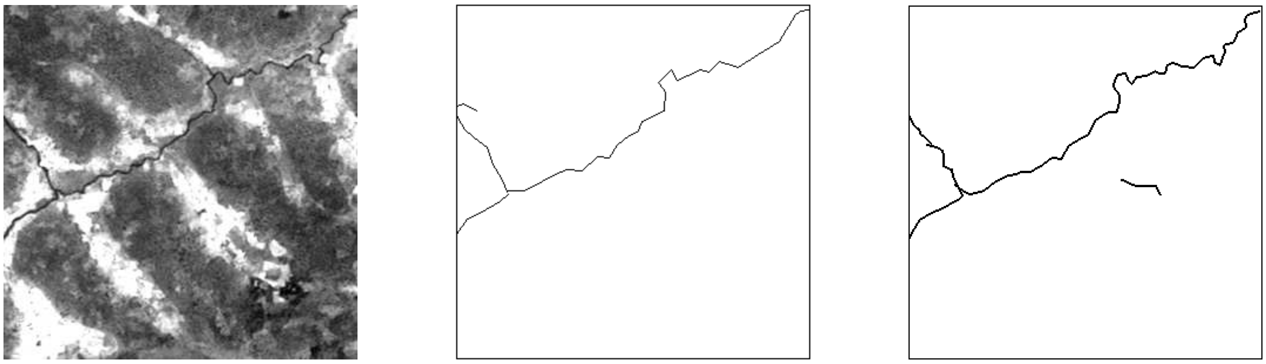
The CAROLINE model improve the detection in terms of accuracy, as shown in figures 12 and 13. Indeed, the data term of our new model allows to **detect sinuous networks with accuracy**. In the case of segments, it is not reasonable to use too small segments as the contrast and homogeneity measure on very small sets of pixels would not be significant. In the case of polylines, we can choose to use mask sections longer than the segment of minimal length L_{min}. It is thus possible to use very small segments. So, we take $L_{min} = 3$ for river detection on the image of 20 meters of resolution shown in figure 12. The data term computation is done on sections of length of 20 pixels.

Moreover, the **junctions detection is better** using a polyline process than using a segment process as shown by figures 11, 13, and 15. This is due to the use of an interaction of connection that favors the connections of each polyline with the rest of the network. Moreover, the connection distance is minimized by an attractive potential. In the "Quality Candy" model, only the connection between two endpoints is taken into account. The



(a) segment process (b) polyline process

Figure 11: Road extraction on an aerial image (figure 10) : (a) using "Quality Candy" process - RED: segments that fit data (negative potential) - BLUE: segments that do not fit the data (positive potential) - GREEN: external part of the segment mask used for the measure of the contrast with the nearby background; (b) using "CAROLINE" process - GREEN: internal part of the polyline mask, BLUE: external part of the polyline mask - RED: polyline connections.



(a) Satellite image © BRGM 300 × 300 pixels

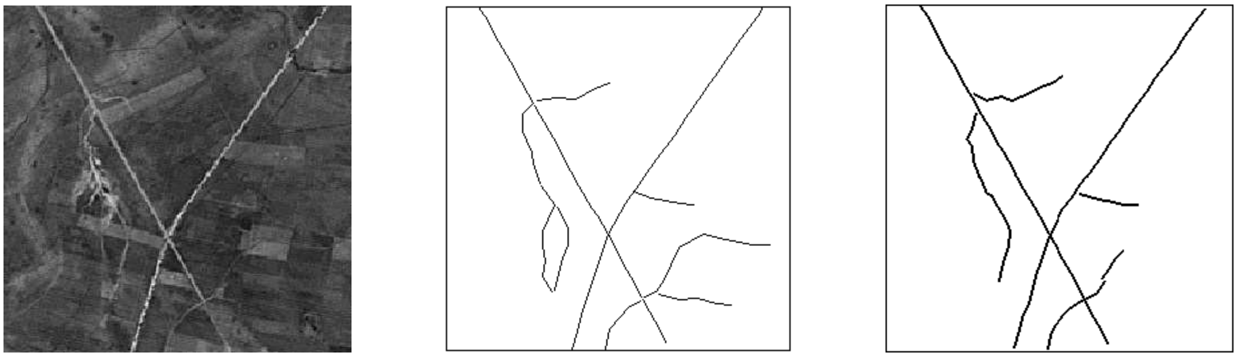
(b) segments obtained in 5 min.

(c) polylines obtained in 45 min.

Figure 12: River extraction from a SPOT XS2 image (20 meters of resolution) provided by the French Geological Survey (BRGM): (b) using the segment process "Quality Candy"; (c) using the polyline process "CAROLINE".

long branches are thus favored but the branch junctions are not favored explicitly. Nevertheless, junctions may be done via the connection with a close extremity. This explains partly the bad quality of the junction between the two sinuous dirt track and a main road in the figure 13. The other reason is that intersections with acute angle ("Y" intersections) are penalized under the law of the "Quality Candy" process.

The proximity interaction introduced in the "Quality Candy" model to penalize the overlapping of segments is not well-defined: it induces a penalization of the "Y" intersections and forbid two segments, whose centers are at a distance lower than the maximal half-length of the two segments, to be quasi-parallel. So, it is impossible to detect two close roads. The **proximity interaction** between two polylines is **better defined**: the "Y" intersections are not penalized and parallel sections are allowed if they are at a distance larger than d. Taking d small, it is thus possible to detect very close roads while forbidding section overlapping. This is illustrated



(a) Satellite Image © CNES
256 × 256 pixels

(b) segments
obtained in 7 min.

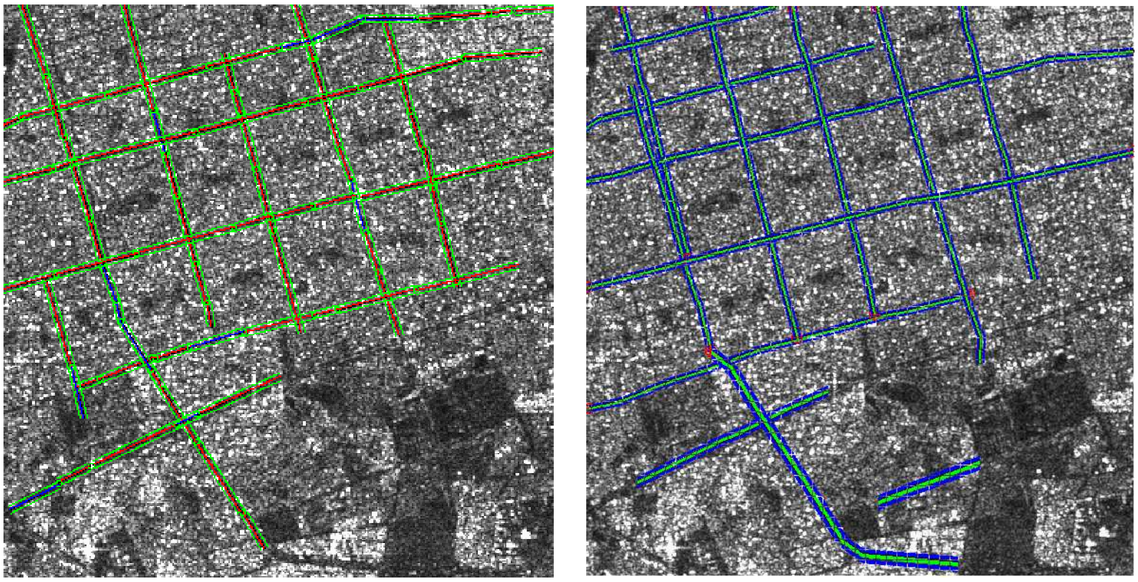
(c) polylines
obtained in 1 h and 15 min.

Figure 13: Road extraction from an image SPOT Panchro (10 meters of resolution) provided by the French Spatial Agency (CNES): (b) using the segment process "Quality Candy"; (c) using the polyline process "CAROLINE".

by the result given in figure 15, where two roads quasi-parallel can be detected using the "CAROLINE" process whereas only one was detected using the "Quality Candy" process.

Figure 14: ERS radar image (525 × 546 pixels) © ESA.

The **computing time** of the polyline process model is nevertheless **important** in comparison to segment process model. This is mainly due to the computation of the data term on the wholeness of the polyline at each proposition of a perturbation, even if the perturbation concerns only one or two segments. The reason is that the data term potential is not based on a sum of potentials for each segment of the polyline but on a uniform decomposition of the polyline mask. Longer the polylines are, longer the computing time will be. So, the cost of an iteration is very high at the end of the algorithm, especially if the roads and rivers present in the image are long and numerous.



(a) segment process (b) polyline process

Figure 15: Road extraction on a radar image (see figure 14) : (a) using "Quality Candy" process - RED: segments that fit data (negative potential) - BLUE: segments that do not fit the data (positive potential) - GREEN: external part of the segment mask used for the measure of the contrast with the nearby background; (b) using "CAROLINE" process - GREEN: internal part of the polyline mask, BLUE: external part of the polyline mask - RED: polyline connections.

5 Conclusion

We have proposed in this report a relevant method to perform line network extraction from satellite and aerial images. This is a fully automatic method without any initialization. Indeed, the use of a simulated annealing scheme using a RJMCMC algorithm for the optimization allows us to initialize the algorithm with the empty configuration. The prior model leads to continuous extracted line networks with few omissions and overdetections. Results on remotely sensed images have shown that the polyline process CAROLINE is fitted to the extraction of sinuous networks and models correctly the road and river junctions thanks to the definition of a polyline connection. Moreover, the data term seems to perform well for different types of images. We will focus in a near future on the definition of an inhomogeneous reference process based on data that would be better adapted to our problem. Indeed, it would allow to accept more correctly positioned segments at the beginning of the algorithm and should decrease the computation time. Moreover, the proposed stochastic modeling allows us to consider working in a frame of data fusion in order to benefit from the contribution of several sources (for instance, multi-sensor or multi-temporal data).

Acknowledgments

The authors would like to thank the French Geological Survey (BRGM) for partial financial support and interesting discussions, the Space Agency (CNES) and the European Space Agency (ESA) for providing the satellite images, the French Mapping Institute (IGN) for providing aerial data.

References

[1] A. Baddeley and M. N. M. van Lieshout. Stochastic geometry models in high-level vision. *Statistics and Images*, 1:233–258, 1993.

[2] M. Barzohar and D. B. Cooper. Automatic finding of main roads in aerial images by using geometric-stochastic models and estimation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 18:707–721, July 1996.

[3] A. Baumgartner, C. Steger, H. Mayer, W. Eckstein, and H. Ebner. Automatic road extraction based on multi-scale, grouping, and context. *Photogrammetric Engineering and Remote Sensing*, 65(7):777–785, July 1999.

[4] U. Bhattacharya and S.K. Parui. An improved backpropagation neural network for detection of road-like features in satellite imagery. *International Journal of Remote Sensing*, 18:3379–3394, April 1997.

[5] I. Couloigner and T. Ranchin. Mapping of urban areas: A multiresolution modeling approach for semi-automatic extraction of streets. *Photogrammetric Engineering and Remote Sensing*, 66(7):867–874, July 2000.

[6] P. Doucette, P. Agouris, A. Stefanidis, and M. Musavi. Self-organized clustering for road extraction in classified imagery. *ISPRS Journal of Photogrammetry and Remote Sensing*, 55:347–358, 2001.

[7] R.O. Duda and P.E Hart. *Pattern Classification and Scene Analysis.* John Wiley & Sons, NY, USA, 1973.

[8] M. A. Fischler, J. M. Tenenbaum, and H. C. Wolf. Detection of roads and linear structures in low-resolution aerial imagery using a multisource knowledge integration technique. *Computer Graphics and Image Processing*, 15:201–223, 1981.

[9] D. Geman and B. Jedynak. An active testing model for tracking roads in satellite images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 18:1–14, 1996.

[10] C. J. Geyer and J. Møller. Simulation and likelihood inference for spatial point process. *Scandinavian Journal of Statistics, Series B*, 21:359–373, 1994.

[11] P.J. Green. Reversible jump Markov chain Monte-Carlo computation and Bayesian model determination. *Biometrika*, 57:97–109, 1995.

[12] A. Grün and H. Li. Road extraction from aerial and satellite images by dynamic programming. *ISPRS Journal of Photogrammetry and Remote Sensing*, 50(4):11–20, August 1995.

[13] D. Haverkamp. Extracting straight road structure in urban environments using IKONOS satellite imagery. *Optical Engineering*, 41(09):2107–2110, September 2002.

[14] C. Lacoste. *Extraction de Réseaux Linéiques à partir d'Images Satellitaires et Aériennes par Processus Ponctuels Marqués.* Phd thesis (in French), University of Nice - Sophia Antipolis, France, September 2004.

[15] C. Lacoste, X. Descombes, and J. Zerubia. Point processes for unsupervised line network extraction in remote sensing. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(10):1568–1579, October 2005.

[16] I. Laptev, T. Lindeberg, W. Eckstein, C. Steger, and A. Baumgartner. Automatic extraction of roads from aerial images based on scale space and snakes. *Machine Vision and Applications*, 12:23–31, 2000.

[17] N. Merlet and J. Zerubia. New prospects in line detection by dynamic programming. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 18(4):426–431, 1996.

[18] W. M. Neuenschwander, P. Fua, L. Iverson, G. Székely, and O. Kubler. Ziplock snakes. *International Journal of Computer Vision*, 25(3):191–201, 1997.

[19] P. H. Peskun. Optimum Monte Carlo sampling using Markov chains. *Biometrika*, 60:607–612, 1973.

[20] M. Rochery, I. H. Jermyn, and J. Zerubia. Higher order active contours and their application to the detection of line networks in satellite imagery. In *Proc. IEEE Workshop Variational, Geometric and Level Set Methods in Computer Vision*, Nice, France, October 2003.

[21] H. Rue and M. Hurn. Bayesian object identification. *Biometrika*, 3:649–660, 1999.

[22] D. Ruelle. Superstable interactions in classical statistical mechanics. *Communication in Mathematical Physics*, 18:127–159, 1970.

[23] M. A. Serendero. *Extraction d'Informations Symboliques en Imagerie SPOT : Réseaux de Communication et Agglomérations.* Phd thesis (in French), University of Nice - Sophia Antipolis, France, December 1989.

[24] R. Stoica, X. Descombes, and J. Zerubia. A Gibbs point process for road extraction in remotely sensed images. *International Journal of Computer Vision*, 57(2):121–136, 2004.

[25] F. Tupin, H. Maitre, J-F. Mangin, J-M. Nicolas, and E. Pechersky. Detection of linear features in SAR images: Application to road network extraction. *IEEE Transactions on Geoscience and Remote Sensing*, 36(2):434–453, 1998.

[26] M.N.M. van Lieshout. Stochastic annealing for nearest-neighbour point processes with application to object recognition. Research Report BS-R9306, CWI, Amsterdam, The Netherlands, 1993.

[27] M.N.M. van Lieshout. *Markov Point Processes and their Applications.* Imperial College Press, 2000.

[28] G. Vosselman and J. de Knecht. Road tracing by profile matching and Kalman filtering. In *Automatic Extraction of Man-Made Objects from Aerial and Space Images*, pages 265–274, Ascona, Switzerland, April 1995.

[29] D. Wang, D.C. He, L. Wang, and D. Morin. Extraction du réseau routier urbain à l'aide d'images SPOT HRV. *International Journal of Remote Sensing*, 17(4):827–833, 1996.

[30] A. Zlotnick and P. Carnine. Finding road seeds in aerial images. *Computer Vision, Graphics, and Image Processing*, 57:243–260, 1993.

Unité de recherche INRIA Sophia Antipolis
2004, route des Lucioles - BP 93 - 06902 Sophia Antipolis Cedex (France)

Unité de recherche INRIA Futurs : Parc Club Orsay Université - ZAC des Vignes
4, rue Jacques Monod - 91893 ORSAY Cedex (France)
Unité de recherche INRIA Lorraine : LORIA, Technopôle de Nancy-Brabois - Campus scientifique
615, rue du Jardin Botanique - BP 101 - 54602 Villers-lès-Nancy Cedex (France)
Unité de recherche INRIA Rennes : IRISA, Campus universitaire de Beaulieu - 35042 Rennes Cedex (France)
Unité de recherche INRIA Rhône-Alpes : 655, avenue de l'Europe - 38334 Montbonnot Saint-Ismier (France)
Unité de recherche INRIA Rocquencourt : Domaine de Voluceau - Rocquencourt - BP 105 - 78153 Le Chesnay Cedex (France)

Éditeur
INRIA - Domaine de Voluceau - Rocquencourt, BP 105 - 78153 Le Chesnay Cedex (France)
http://www.inria.fr
ISSN 0249-6399