Estimating a CBRN atmospheric release in a complex environment using Gaussian Processes

Adrien Ickowicz, François Septier, Patrick Armand

Estimating a CBRN atmospheric release in a complex environment using Gaussian Processes

Adrien Ickowicz*, François Septier†, and Patrick Armand‡
*Université de Lille1 / LAGIS UMR CNRS 8219
Cité Scientifique, Villeneuve d'Ascq Cedex, France
Email: adrien.ickowicz@telecom-lille1.eu
†Institut Mines-Télécom / Télécom Lille1 / LAGIS UMR CNRS 8219
Cité Scientifique, Villeneuve d'Ascq Cedex, France
Email: francois.septier@telecom-lille1.eu
‡CEA, DAM, DIF
Arpajon, France

***Abstract*—In this paper, we present a new methodology for the estimation and the prediction of the concentration of pollutant in a complex environment. We take benefit of a semi-parametric formulation of the problem to perform a faster and more efficient estimation of the pollutant cloud. In a first part, we present how we use the Gaussian process to model the interactions between position and time given the observations. Then, we introduce the expansion as a function of the observations through the time, and we construct an estimator of the time of release from it within change-point detection framework. Then, we use this time estimate to obtain the position (or more likely, a confidence region of the position) of the source. Several simulations are provided in a complex city scenario that demonstrate the accuracy of the proposed technique.**

I. Introduction

A Lagrangian stochastic model describes the paths of fluid particles in a turbulent flow, given a knowledge (i.e., statistical description) of the random velocity field. It is the natural and the most powerful mean to describe many interesting atmospheric processes (e.g., the dispersion of pollen, or of air pollutants), and with the aid of such models we can expect eventually to develop better strategies for, as an example, the application of aerial sprays.

However, depending on the complexity of the environment in which the fluid particles evolve, the Lagrangian model suffers form its complexity. It may take an incredible amount of time and computer resources to be able to reconstruct the pollutant cloud. This modelling is then particularly inaccurate when the point of the study is to use the simulated cloud as a tool, for example when one wants to reconstruct the pollutant source given a sample of concentration observations. So far, multiple attempts to minimize the estimation time have been tried, mainly towards the optimization of the estimation algorithm. We explore in this paper another way to deal with that issue, starting with a learning-based technique to model the atmospheric dispersion.

There exists several works on the application of the Gaussian processes for the modelling of environmental phenomena. Among them, Graepel [2] defines a methodology when one may describe the natural phenomena with partial differential equations (pde). More recently, Sarkka [1] or Xu et al [3], [4] modelled stochastic pde's and time dependant pde's of one particle trajectory. However, the kernel involved in these models remained very simple, and no studies have been tried so far to model the concentration evolution of a great amount of fluid particles which trajectories are defined by stochastic pdes. Therefore, we propose in this work to fill this gap by considering a drift-dependant kernel. Such modifications will allow us to express more sophisticated beliefs about the structure of the concentration.

The paper is divided into six main parts. After this introduction, we present the pollutant release issue, and the complex environment we have to deal with. We will underline the purpose of our work, which will leads us toward the next section on Gaussian process. After a brief recall on what can be done with Gaussian processes, we present our methodology and the construction of the kernel that is of the main contribution in this proposed learning method. On the fourth part, we introduce a function of the observations and the sensors, called expansion, and we present the main properties of it. Following this part, we present the source parameters estimators and the techniques used to perform the estimation. We finish with a bench of simulation results in a complex scenario. We demonstrate that the cloud is well reconstructed. The source term estimation provided is proven accurate despite the small amount of sensors we have to deal with.

II. The dispersion model and the complex environment

A. Dispersion model

There exists a large number of models in the literature, based on various hypothesis, including varying boundaries constraints, for example. Anyway, according to Wilson and Sawford [6], a general expression can be extract though. With C being the concentration of hazardous material, u the known wind velocity field, K the eddy-diffusion coefficient, and Q

the source function, we have:

$$\begin{aligned} \frac{\partial C}{\partial t} + u\nabla C - \nabla(K\nabla C) &= Q \\ \text{s.t. } \nabla_n C = 0 \text{ at } \partial\Omega & \end{aligned} \quad (1)$$

Moreover $Q = q_s\delta(x-x_0)[H(t-t_{on}) - H(t-t_{off})]$, with H the Heaviside unit step function. Under certain hypothesis on Ω and $\partial\Omega$, there also exists a backward stochastic differential equation such that:

$$\begin{aligned} -\frac{\partial C^*}{\partial t} - u\nabla C^* - \nabla(K\nabla C^*) &= h \\ \text{s.t. } K\nabla_n C^* + u_n C^* = 0 \text{ at } \partial\Omega & \end{aligned} \quad (2)$$

where h is the detector response function. This concentration formulation is based on the stochastic evolution of a pollutant release in the atmosphere. The initial modelling for one particle is formulated with a system of stochastic differential equations such that:

$$\begin{aligned} dX_t &= U_t dt \\ dU_t &= a(X_t, U_t, t)dt + (C_0\epsilon(X_t, t))^{1/2} dW_t \end{aligned} \quad (3)$$

where X and U correspond to the position and velocity of a marked fluid particle. $a(.)$ represents the drift coefficient vector, $\epsilon(.)$ the volatility and C_0 the Kolmogorov universal constant. This Lagrangian Stochastic model will be used for the simulation of the marked fluid particles needed to obtain the mean concentration of pollutant.

B. The observations

We may assume that we have noisy observations by adding a random variable ζ, such that:

$$R_{t_j}^{obs(i)} = D_{t_j}^{true(i)}(\theta) + \zeta_{t_j}^i \quad (4)$$

The noisy term ζ is assumed to have a derivable probability function F_ζ on the set of interest, mean 0 and finite variance. The last step before the estimation of θ is the determination of the expression of D_t^{true}. These values correspond to the mean concentration "seen" by the sensors and that can be expressed by:

$$D_{t_j}^{true(i)}(\theta) = \int_0^T \int_\Omega C(x,t)h(x,t|x_i,t_j)dxdt \quad (5)$$

where C is the previously defined concentration function, and h the filter function associated to the sensor i, for a measurement at time t_j. As long as C depends on θ, the writing of Eq.4 takes its justification. The calculation of this integral may cause some difficulties and ask for computer resources. Using Eq. 2, this $(dim(\Omega)+1)$-uple integral can be reduce to dimension 1, so that:

$$\begin{aligned} D_{t_j}^{true(i)}(\theta) &= \int_0^T \int_\Omega C^*(x,t)Q(x,t|x_i,t_j)dxdt \\ &= q_s \int_{t_{on}}^{t_{off}} C^*(x_0,t)dt \end{aligned} \quad (6)$$

In the next parts of the paper, we will consider that the measurements at the sensors positions and time are equal to the concentration at the sensors position and time (which means that $h(.,.) = \delta(x_i, t_j)$).
The next section is centred on the Gaussian processes, and their use in our estimation problem.

III. The Gaussian process ($\mathcal{GP}$)

A. Definition

A Gaussian Process is a set of random variable f_x indexed by $x \in \mathbf{R}^d$ and taking values on $\mathbf{R}$ (for the sake of simplicity we will note $f(x)$). Then, considering a collection of input values $\mathbf{x} = \{x_1, ..., x_n\}$, $\mathbf{f} = [f(x_1), ..., f(x_n)]^T$ is distributing according to a multivariate Gaussian distribution. Therefore, a $\mathcal{GP}$ is completely specified by its two first moments, the mean

$$m(\mathbf{x}) = \mathbb{E}[f(\mathbf{x})] \quad (7)$$

and a covariance function

$$\kappa_{1:n} = \mathbb{E}\big[(f(\mathbf{x}_{1:n}) - m(\mathbf{x}_{1:n}))(f(\mathbf{x}_{1:n}) - m(\mathbf{x}_{1:n}))^T\big] \quad (8)$$

One may see the $\mathcal{GP}$ as a non-parametric estimation with stochastic weights on the observations. Indeed, this implies that it suffers form the curse of dimensionality.
Therefore, the random function form at any spatial and temporal location of the concentration will be considered as distributed according to a $\mathcal{GP}$ as follows

$$C(.) = \mathcal{GP}(m(.), \kappa(.,.)) \quad (9)$$

A common prior specification about the Gaussian process is to consider the mean to be null over the set. Considering this, and given the measurements, the posterior distribution of the $\mathcal{GP}$ may be written:

$$\begin{aligned} \mathcal{L}\big(\mathbf{f}^*|\mathbf{x}^*, \mathbf{x}, \mathbf{f}\big) = \mathcal{N}\big(&\kappa(\mathbf{x}^*, \mathbf{x})\kappa(\mathbf{x}, \mathbf{x})^{-1}\mathbf{f}, \\ &\kappa(\mathbf{x}^*, \mathbf{x}^*) - \kappa(\mathbf{x}^*, \mathbf{x})\kappa(\mathbf{x}, \mathbf{x})^{-1}\kappa(\mathbf{x}, \mathbf{x}^*)\big) \end{aligned} \quad (10)$$

Our main assumption here is to consider that the concentration can be modelled as a $\mathcal{GP}$. Therefore, in all the previous equations, $\mathbf{f}$ will stand for the concentration values, and $\mathbf{x}$ for the location of the sensors, and the time of measurements. The predicted concentration will be denoted $\mathbf{f}^*$, and $\mathbf{x}^*$ stands for the location (time and space) where we want that concentration to be predicted.
Considering this, we had to make a choice on the expression of the kernel. A decided to compare a common used Kernel, called Isotropic Kernel, with a proposed drift-dependant Kernel.

B. The Isotropic Kernel

In the classical literature on Gaussian Process, the Isotropic kernel has clearly the most important use over the models. The reason is that we are required to specify a function which will generate a non-negative definite covariance matrix for any set of point. The idea is to specify covariances so that points close

to inputs will give similar predictions. The expression of this kernel is:

$$\kappa_{iso}(\mathbf{x},\mathbf{x}') \quad = \quad \frac{1}{\alpha}\exp\Big(-\frac{\|\mathbf{x}-\mathbf{x}'\|^2}{\beta^2}\Big) \qquad (11)$$

where α and β are the hyper-parameters to be estimated.

C. The drift-dependant kernel

The most important part of the learning process is based on the choice of the kernel for the Gaussian process. In order to get the most accurate estimation, the kernel has to represent as precisely as possible the interactions between two concentration at two different positions and times, without being two much specific. Indeed, with too much precision, the observations would be neglected, which is not the purpose of this learning technique. Given the spatial and time dependency of the dynamic, the accurate kernel will have to consider both this spatial and time dimension. So far, classical exponential kernels that dealt with this issue where constructed as a product across time and position, or combined kernel [3].
What we propose in this article is the use of a more sophisticated kernel that will allow us to perform a better estimation, at the expense of being more complex. Indeed, to built that kernel, we consider that the concentration evolution is based on three steps:

- Generate a wind field over the complex environment;
- Consider reflection on the buildings;
- Generate the random walk of the particles.

The complexity of the scenario is based on the three previous items. However, the reason for which it is time demanding to simulate a such scenario is the number of particles that have to be generated with respect to this three conditions, especially the reflection step and stochastic step. We consider however that the main part of one particle behaviour is due to the wind field. We then consider that a wind field-based kernel is appropriate to reflect the relationship between two points.
The main difficulty with that dynamic is the difficulty to write it with a closed form expression. We recall that u being the *known* wind velocity field, it depends on both x and t so that:

$$\begin{array}{rccc} u: & \Omega\times[0,T] & \rightarrow & \Omega \\ & (x,t) & \rightarrow & u(x,t) \end{array} \qquad (12)$$

Then, the trajectory of a particle starting at point x_0 at time t_0 is the solution of the following system:

$$\left\{\begin{array}{rcl} \dot{x} & = & u(x,t) \\ x(t_0) & = & x_0 \end{array}\right. \qquad (13)$$

Let note $s_{x_0,t_0}(t)$ the solution of this system. We may say that s is a trivial solution when u is constant (in which case: $s_{x_0,t_0}(t) = \mathbf{u}\times(t-t_0)+x_0$), or does not depend on x (then we have: $s_{x_0,t_0}(t) = \int_{t_0}^{t} u(t)dt + x_0$). The existence and uniqueness of the solution being well defined, we then define the following kernel (with $\mathbf{x} = (x,t)$):

$$\kappa_{dyn}(\mathbf{x},\mathbf{x}') \quad = \quad \frac{1}{\sigma(t,t')}\exp\Big(-\frac{d_s(\mathbf{x},\mathbf{x}')}{2\sigma(t,t')^2}\Big) \qquad (14)$$

where we have:

$$\left\{\begin{array}{rcl} d_s(\mathbf{x},\mathbf{x}') & = & (x'-s_{x,t}(t'))^2+(x-s_{x',t'}(t))^2 \\ \sigma(t,t') & = & \alpha\times|t-t'|^{\beta}+1 \end{array}\right.$$

where we assume that s corresponds to the solution of the previous system. As denoted in Rasmussen et al [7], (α,β) is the *characteristic length-scale*. This kernel is clearly symmetric, its value is equal to 1 for the same point at the same time, it only depends on the distance when the time is identical, and on the time when the location remains the same. Physically speaking, this kernel seems to be a promising candidate. Although some improvements have been made in the calculation of the estimated hyper-parameters of the Gaussian process, like the ICM algorithm [5], the maximum likelihood technique will grant the results in our situation without any trouble.

IV. Cloud expansion

The reconstruction of the pollutant cloud is a mean to estimate the source parameters. Indeed, we assume that a good reconstruction will lead to a good estimation of the parameters. Then, a first approach to estimate the source parameters would be to detect the maximum of concentration over space and time. This maximum would stand for the strength of the source, and the location in space and time of this maximum would give us the source location and release time. The problem with this approach is that the maximum of the reconstructed cloud will be equal to an observation according to the Gaussian Process formulation. Then, the maximum will be located on a sensor, which has very low chance to be the real location of the release.
Our approach is based on the phenomenon observation that the concentration of CBRN material is located in a very small area at the beginning of the release, and then spread all over the area of interest. Then, we define the expansion of the release as a function which is minimum at time of release, and then increase over time. This expansion can be expressed:

$$E(t) \quad = \quad \int_{\Omega} C(x,t)\Big(x-\int_{\Omega} yC(y,t)dy\Big)^2 dx \qquad (15)$$

$E(.)$ is clearly null until the release is made, and then becomes positive until all the concentration over the space of interest vanished.
The first problem encountered is that we do not have the concentration values, but some observations of it. Then the expansion is calculated with f instead of C, where $f = C+\varepsilon$. ε is a combination of the noise of the sensor measurements and the concentration background of the scenario. Then, $\mathbb{E}[\varepsilon] = b \in \mathbf{R}_+$ and $\mathbb{V}[\varepsilon] = \sigma^2 \in \mathbf{R}_+$. The observed expansion

becomes stochastic, calculated with the observations and may be written:

$$\mathbf{E}^n(t) = \sum_{i=1}^{n} f(x_i,t)/\bar{f}(t)\Big(x_i - \sum_{j=1}^{n} x_j f(x_j,t)/\bar{f}(t)\Big)^2 \quad (16)$$

where $\bar{f}(t) = \sum_{i=1}^{n} f(x_i,t)$. This noisy version may be simplified if we consider that the noise is smaller than the concentration after t_0, for the sensors located in the "path" of the cloud. If we denote,

$$E^n(t) = \sum_{i=1}^{n} \frac{C(x_i,t)}{\bar{f}(t)} x_i^2 - \Big(\sum_{i=1}^{n} \frac{C(x_i,t)}{\bar{f}(t)} x_i\Big)^2 \quad (17)$$

under certain condition on the sensors location (homogeneous repartition over the set), we obtain that:

$$E^n(t) \xrightarrow[n\to+\infty]{} E(t) \quad (18)$$

And we may write:

$$\mathbf{E}^n(t) = \begin{cases} E^n(t) + \nu(t) - \eta(t)^2 & \text{if } t < t_0 \\ E^n(t) - 2\eta(t)m^n(t) + \nu(t) - \eta(t)^2 & \text{if } t \geq t_0 \end{cases} \quad (19)$$

where

$$\eta(t) = \sum_{i=1}^{n} \varepsilon(x_i,t)/\bar{f}(t) x_i$$
$$\nu(t) = \sum_{i=1}^{n} \varepsilon(x_i,t)/\bar{f}(t) x_i^2 \quad (20)$$

are random variables and $m^n(t) = \sum_{i=1}^{n} C(x_i,t)/\bar{f}(t) x_i$. The two random variables have different behaviour before and after the time of release. A representation of the expansion is presented in Fig. 1. The time of release is located at $t = 150$ (or $t = 2:30$), and we calculate the real expansion based on the concentration value on every point of the set. This representation is however the result of a numerical approximation of the Eq. 15, with $\Delta x = 0.1$. Before this release time, the expansion value is not null because we assume that a background concentration exists, which value is not null.

One may observe that the expansion is expected to raise to 0 at the moment of the release, and then increase continuously until the concentration over the space of interest vanishes. So, to estimate the time of release we estimate the time when the expansion has its smallest value. However, due to the random variables presented in Eq. 19, the minimum won't necessary be reached at the right time. The solution we choose was to estimate the expansion function and classify it in two steps, the first one when constant, and then the second step when the expansion is increasing. The methodology is presented in the next subsection.

The low number of sensor maybe a great limitation to the quality of estimation of the time of release. An alternative approach for calculating the expansion is to run a Gaussian Process to reconstruct the cloud over space and time and use it to calculate the expansion:

$$\mathbf{E}^n(t) = \sum_{i=1}^{n_t} \hat{f}(x_i,t)\Big(x_i - \sum_{j=1}^{n_t} x_j \hat{f}(x_j,t)\Big)^2 \quad (21)$$

where n_t stands for the number of points considered in the reconstruction. This methodology is longer than the previous one, given that we have to reconstruct the CBRN cloud over the whole time of interest, and not only on the observed time. However, with a good knowledge of the underlying dynamic, one may expect a greater accuracy on the estimation of the expansion function, and therefore on the time of release. Once the time is estimated, we use our Gaussian Process reconstruction of the cloud at that time to perform an estimation of the source position by simply finding the location of the maximum. This estimator will strongly depend on the quality of the time estimation as well as the accuracy of the Gaussian process reconstruction.

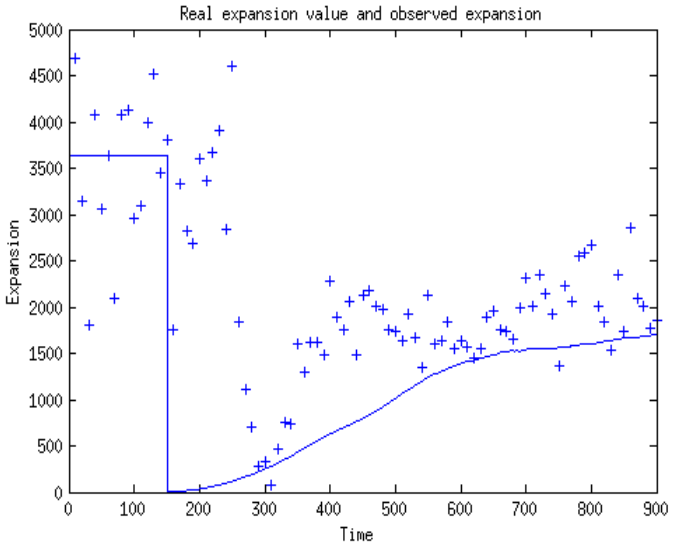

Fig. 1. Example of expansion evolution. The straight line represents the numerical approximation of Eq. 15, and the crosses represent the noisy expansion.

V. Estimating the source parameters

We assume that we have a set of concentration measurements $\mathbf{C_n}$, at several locations (space and time) $\mathbf{x_n}$.

A. Expansion estimation

The key point for a good estimation is a good understanding of the underlying problem. The main point here is to provide a correct estimation of the time of release, which corresponds to the time when the expansion has its minimum value. Theoretically, this value should be null. However, the background noise and the non-continuous status of the observations make it strictly positive. Several approaches can be taken for the purpose.

As we stated before, their exist three phases for the expansion. First it is almost constant, then there is a jump that raise

it through 0, and then it increases. In a theoretical case when measurements occurs all over the space of interest, the jump phase duration is equal to 0. This is most obvious if we take the logarithm of the expansion, represented in Fig. 2. Considering it, we is a smart move to consider that

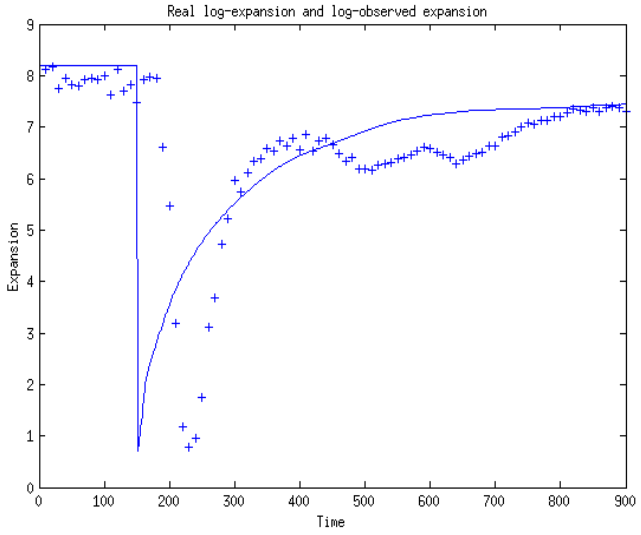

Fig. 2. Example of log-expansion evolution. Same representation as Fig. 1, with a log scale.

the minimum value of the expansion will lead to a good estimation of the time of release. However, it would be a biased estimator, depending on the proximity of sensors to the source location.

Therefore, we will focus our effort in the construction of a classification estimator that will allow the estimation of the change-point, as it is shown in Fig. 3. According to Arlot

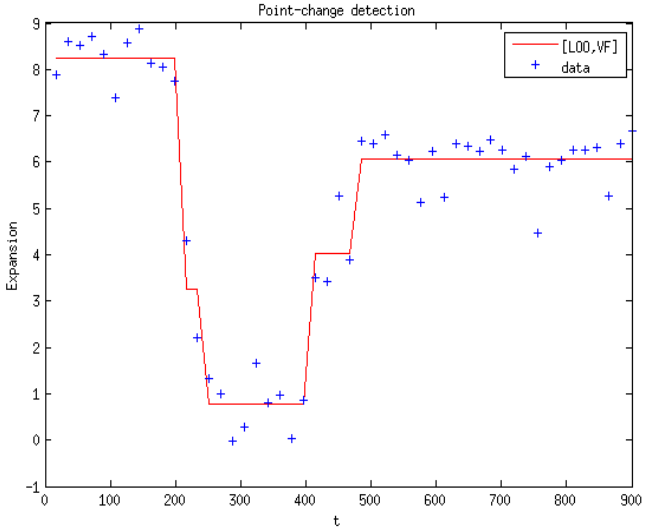

Fig. 3. Example of point-change detection. Calculated expansion is represented with the blue crosses, the red lines represent the estimated steps with the Leave one out methodology.

[8], we test four statistical techniques and one heuristic for the estimation of point-change. We compare a penalized maximum likelihood algorithm (denoted PML), three empirical risk minimization-based algorithms (denoted LOO, ERM-VF, ERM-BM), and an estimator based on the minimum of the expansion function (denoted min_d).

a) [PML] Penalized maximum likelihood estimation: PML aims at detecting changes in either the mean or the variance.

b) [LOO] Leave-one-out empirical risk minimization: It consists in training with the whole sample except one point, used for testing, and repeating this for each data point.

c) [ERM-VF] V-fold empirical risk minimization: The idea is to first partition the data into V blocks, to use all the data but one block as a training sample, and to repeat the process V times. In other words, ERM-VF is a block-wise Leave-one-out.

d) [ERM-BM] Birgé-Massard empirical risk minimization: The idea here is to minimize a criterion that has been proven [9] to be an unbiased estimator of the quadratic loss when the data are homoscedastic.

e) [min_d] Minimum log-expansion estimation: This simple estimator is based on the calculation of the minimum of the expansion upon time. Then, the estimation of the time release will be biased (depending on the distance between the sensors and the source).

B. Position estimation

In the two-stage procedure for the estimation of the source parameters, one can figure that having a good estimation of the time of release is important, but having an accurate construction of the pollutant cloud is important too. Then, if we note $\hat{t}_s$ the estimated time and $\mathbf{x_s} = (\Omega, \hat{t}_s)$ the inputs to be predicted, and according to the section III-A, the reconstructed cloud will have the following expression:

$$\forall x \in \Omega, \quad \hat{C}(x, \hat{t}_s) \quad = \quad \mathbf{f}(\mathbf{x_s}) \tag{22}$$

where,

$$\begin{aligned} \mathbf{f}(\mathbf{x_s}) \quad \sim \quad & \mathcal{N}\big(\kappa(\mathbf{x_s}, \mathbf{x_n})\kappa(\mathbf{x_n}, \mathbf{x_n})^{-1}\mathbf{C_n}, \\ & \kappa(\mathbf{x_s}, \mathbf{x_s}) - \kappa(\mathbf{x_s}, \mathbf{x_n})\kappa(\mathbf{x_n}, \mathbf{x_n})^{-1}\kappa(\mathbf{x_n}, \mathbf{x_s})\big) \end{aligned} \tag{23}$$

According to these equations, the predicted cloud is stochastic. Anyhow, the proposed estimator of the source location is defined as follows:

$$\hat{x}_s \quad = \quad \underset{x \in \Omega}{\operatorname{argmax}} \quad \mathbb{E}\big[\mathbf{f}(\mathbf{x_s})\big] \tag{24}$$

and we have for final estimator:

$$\hat{\mathbf{x}}_\mathbf{s} \quad = \quad \big(\hat{x}_s, \hat{t}_s\big) \tag{25}$$

One should also notice that given the distribution of the Gaussian process, the maximum of the expectation correspond to the maximum of the posterior mean.
We present in Fig. 4 a scheme of our estimation procedure.

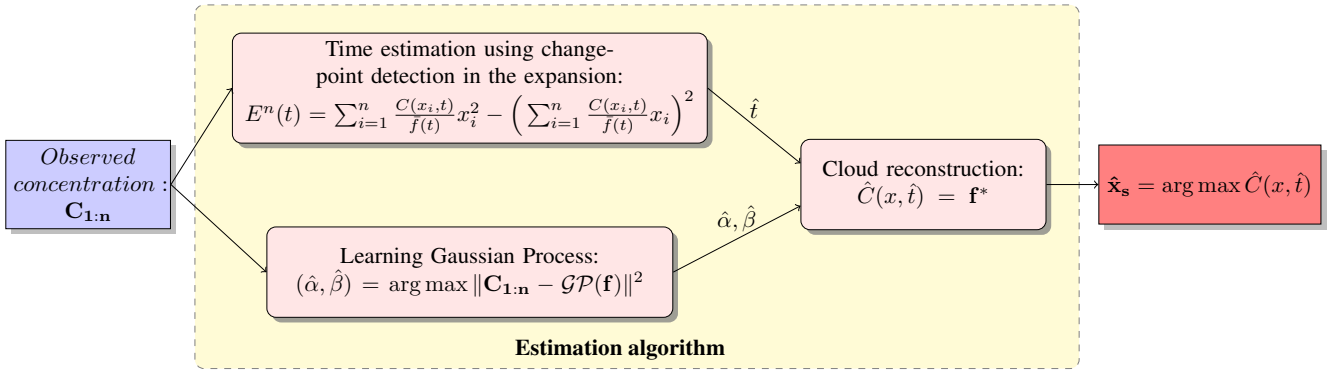

Fig. 4. Scheme of the estimation procedure.

VI. Simulation results

To test the quality of the proposed methodology, we study the results provided in the complex scenario of a release in a city. The map of the city is presented in Fig. 5. Given that the proposed procedure is two-step based, we first study the quality of the time-of-release estimation with different change-point detection algorithms, and then we provide a complete estimation of the source position and time of release.

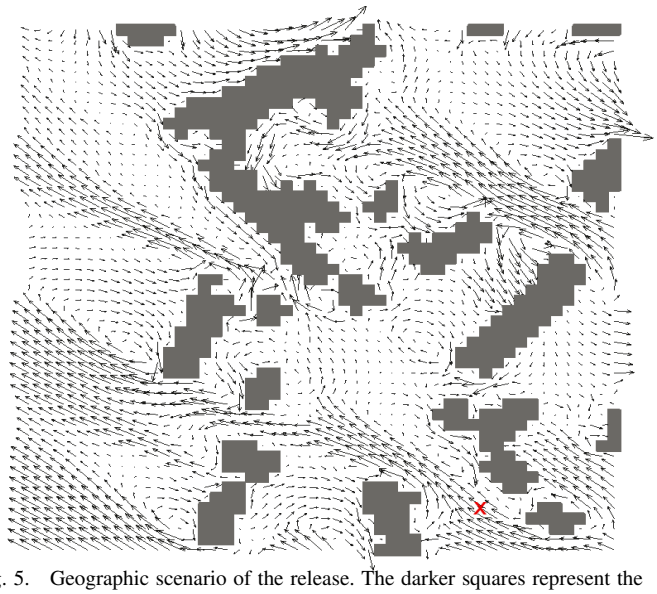

Fig. 5. Geographic scenario of the release. The darker squares represent the buildings, while the wind velocity is represented with the black arrows. The red cross stands for the source location.

A. Change-point detection

To test the detection algorithms, we simulated a release at time $t_0 = 150$, for a source located in the complex scenario presented in Fig. 5 at $(x_0 = 115, y_0 = 10)$. A network of 5 sensors is placed over the space of interest, and provide time-continuous observations on the CBRN concentration. The comparison between the five techniques is presented in Table I, the mean, standard deviance and empirical confidence interval having been calculated through 500 scenarios.

	Mean	Std Dev.	Empirical 0.95 CI
min_d	3:22	2:42	[0:54, 9:27]
LOO	5:32	6:08	[1:03, 15:00]
ERM-VF	5:22	3:57	[1:12, 15:00]
ERM-BM	5:34	6:00	[2:15, 15:00]
PML	**2:57**	**1:42**	**[1:06, 6:03]**

TABLE I
Comparison between the point-change estimators with 5 sensors.

It appears that from all the proposed techniques, the heuristic estimator and the penalized maximum likelihood method have the better results when the number of sensors is very low. If a bias is observed form the heuristic estimation, the PML has no such trouble. It comes certainly form the greater sensibility of this methodology to small changes in the signal. Therefore, earlier changes are detected in the estimation procedure, leading to smaller than expected time of release estimators.
To analyse the influence of the number of sensors, we compare the estimators of each method for an increasing number of sensors, starting with 5, and increasing it through 50. As one can see on Fig. 6, the PML method provides a good estimation of the time of release even with a low number of sensors. The two other presented estimators have a bias that decreases toward 0 as the number of sensor increase.

B. Isotropic Kernel estimation

Using the PML technique, we try to estimate the source location by reconstructing the CBRN cloud with the Gaussian processes. We first provide an estimation calculated with a classical isotropic kernel involving both the space and the time. The results are presented in Table II. The main issue encountered here is obviously the fact that with a low number of sensors, the source location is nowhere close to a sensor (or with a very small probability). Given that the Gaussian process estimation is highly reliable in the neighbourhood of

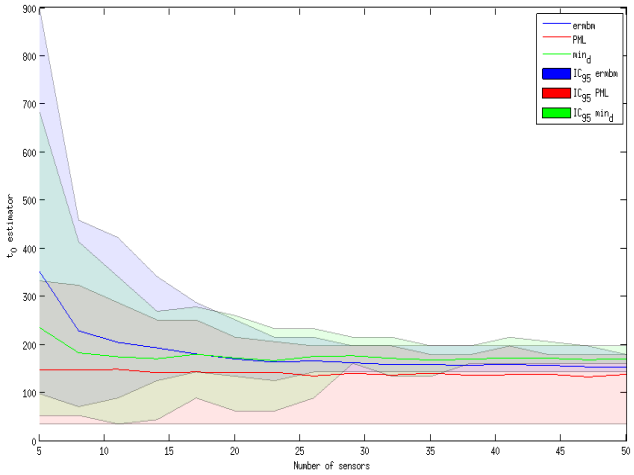

Fig. 6. Value of the estimator of t_0 depending on the number of sensors

	♯	$\hat{x_0}$	$\hat{y_0}$	$\sigma(\hat{x_0})$	$\sigma(\hat{y_0})$
min_d	5	68.97	62.58	42.82	38.96
	20	**97.13**	**26.37**	**27.64**	**26.08**
	50	**104.47**	**21.60**	**28.94**	**19.47**
PML	5	81.90	65.37	41.52	40.19
	20	85.40	47.80	40.77	43.51
	50	94.77	37.01	42.00	40.25
ERM-BM	5	69.54	54.78	44.13	36.08
	20	83.54	42.24	39.56	41.46
	50	96.66	34.70	39.13	38.79
LOO	5	**69.21**	**54.24**	**44.02**	**35.83**
	20	82.61	47.91	42.35	43.58
	50	96.28	36.74	40.95	39.23

TABLE II
ESTIMATION WITH THE ISOTROPIC KERNEL. COMPARISON BETWEEN THE ESTIMATORS (5, 20 AND 50 SENSORS).

the measurements, and quite of lower confidence when it is not, it is not surprising that the position estimation with a simple isotropic kernel is that bad for a low number of sensors. However, Table II validates the methodology when the number of sensors increases. Indeed, for 50 sensors, the estimation of the position is very correct, even with an isotropic kernel.

C. Drift-dependant Kernel estimation

In order to overcome the limitation of the isotropic kernel within the Gaussian process estimation with a low number of sensors, we performed the simulations with the dynamic kernel presented in section III-C. The conditions of simulation are the same, and we compare the position estimation given four different time estimation methods. This is done in order to obtain the most accurate methodology. Indeed, having a best estimation on the time will not necessarily lead to better position estimation.

From Table III, we can observe that the use of this drift-dependant kernel improves quite significantly the performance of the source localisation, at the expense of more computation. Indeed, computing the covariance with dynamic kernel is longer than computing the covariance with isotropic kernel. However, it remains very faster than a classical MCMC estimation, even with an adaptive algorithm. In Fig. 7, a comparison between the real concentration cloud and the ones reconstructed with the Gaussian process based on either the isotropic kernel and the drift-dependant kernel is depicted. These reconstructions of the cloud at different times clearly show the superiority of the drift-dependant kernel which allows to capture more complex correlation structure in space and time of the concentration.

	♯	$\hat{x_0}$	$\hat{y_0}$	$\sigma(\hat{x_0})$	$\sigma(\hat{y_0})$
min_d	5	**108.94**	**12.21**	**42.00**	**17.05**
	20	120.28	5.28	12.50	4.64
	50	**114.51**	**6.48**	**6.37**	**3.07**
PML	5	97.69	18.65	30.91	13.65
	20	110.37	9.53	7.32	6.30
	50	113.44	6.56	3.23	3.23
ERM-BM	5	84.03	24.33	38.24	16.83
	20	**111.32**	**9.15**	**7.73**	**5.94**
	50	113.36	6.64	3.26	3.26
LOO	5	84.23	24.52	37.09	16.51
	20	110.28	9.81	6.54	6.28
	50	113.31	6.65	3.26	3.28

TABLE III
ESTIMATION WITH THE DRIFT-DEPENDANT KERNEL. COMPARISON BETWEEN THE ESTIMATORS (5, 20 AND 50 SENSORS).

VII. CONCLUSION

We present in this article a new and faster method for the estimation of source parameters when a release of CBRN materials occurs. By defining the expansion, we first construct an estimator of the time of release based on the change-point detection. The comparison between several methodologies demonstrate the accuracy of the proposing methodologies despite the bias, even with a low number of sensors through the field of interest. Then based on the wind velocity field, we construct a Gaussian process estimation of the concentration evolution. The use of this reconstruction allows to built an accurate estimator of the source position. The results presented in the simulation section underlined the strengths of the method. First, it is faster than a classical MCMC estimation, and still provide a good estimation. The Gaussian process approach allows not only the estimation of the parameters, but also the possibility to track the contaminant cloud. Moreover, the drift-dependant kernel proposed proved to be robust despite the low number of sensors provided.

This approach leaves several challenges. The expansion expression allows the estimation of a biased time of release, and should certainly contain the information of the real time of release, if linked with the dispersion equations. In addition,

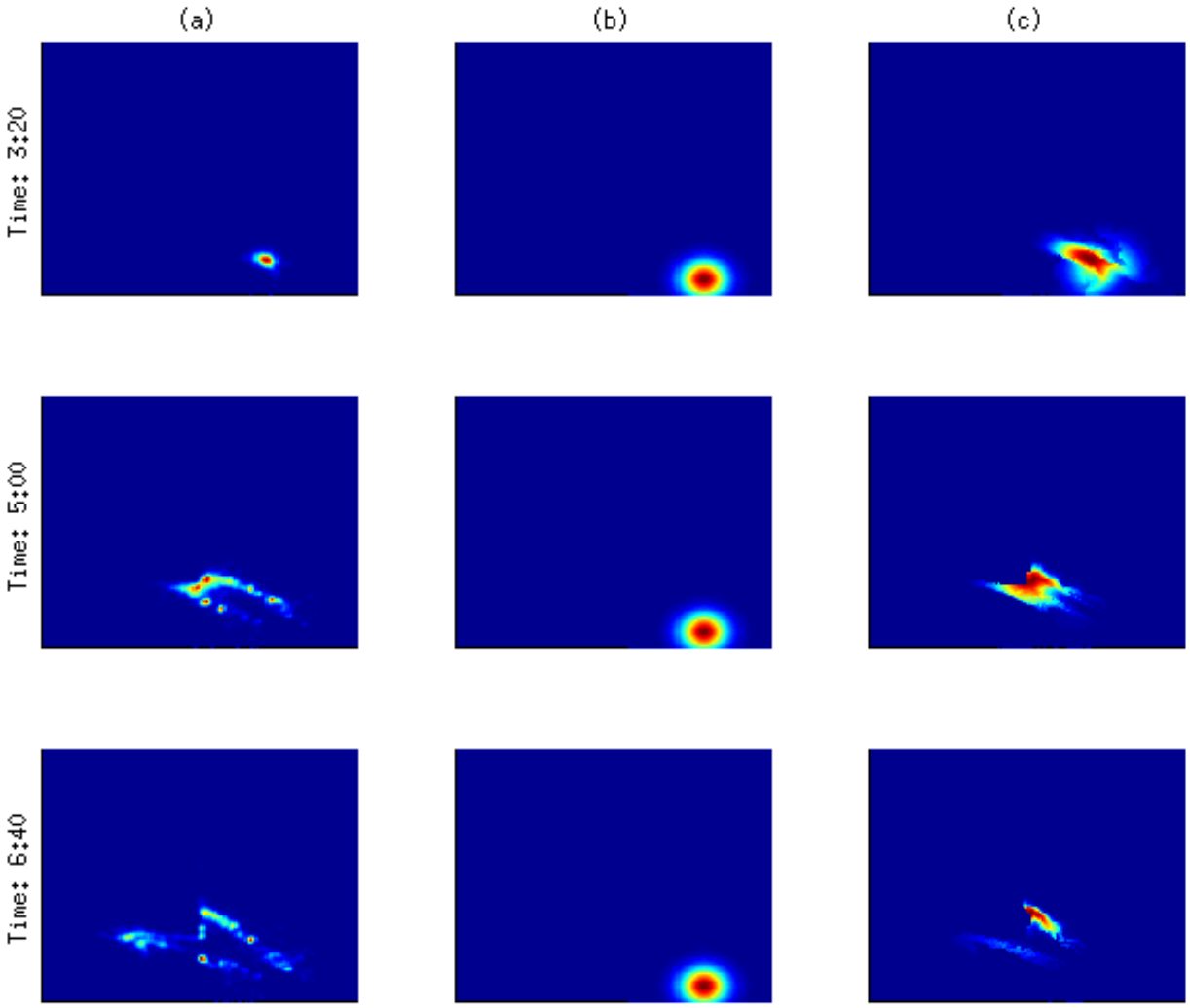

Fig. 7. Comparison between the real concentration cloud and reconstruction based on the Gaussian Process at three different times. The red color represents the highest concentration value, the blue represents an almost null value. (a) Particle cloud concentration (b) Cloud reconstruction with Isotropic Kernel (c) Cloud reconstruction with drift-dependant Kernel

this Gaussian process approach, while fast, remain quite long given the need to calculate the drift-dependant kernel based on the dispersion. It should be interesting to study the accuracy of a *lighter* kernel.

ACKNOWLEDGMENT

The authors would like to thank the CEA and the GIS 3'SGS for their support and funding.

REFERENCES

[1] S. Sarkka, *Linear Operators and Stochastic Partial Differential Equations in Gaussian Process Regression*, Proceedings of International Conference on Artificial Neural Networks (ICANN), 2011.

[2] T. Graepel, *Solving noisy linear operator equations by Gaussian processes: Application to ordinary and partial differential equations*, Machine Learning-International Workshop then Conference, 2003.

[3] Y. Xu and J. Choi, *Adaptive Sampling for Learning Gaussian Processes Using Mobile Sensor Networks*, Sensors 2011, 11, 3051-3066.

[4] Y. Xu, J. Choi, S. Dass and T. Maiti, *Sequential Bayesian Prediction and AdaptiveSampling Algorithms for Mobile Sensor Networks*, under revision for IEEE Transactions on Automatic Control.

[5] I. Nevat, G. W. Peters and I. B. Collings, *Location-aware Cooperative Spectrum Sensing via Gaussian Processes*, submitted

[6] J. D. Wilson and B. L. Sawford, *Review of Lagrangian stochastic models for trajectories in the turbulent atmosphere*, Boundary-Layer Meteorology, 87,191–210,1996

[7] C. E. Rasmussen and C. K. I. Williams, *Gaussian Processes for Machine Learning*, the MIT Press, 2006.

[8] S. Arlot and A. Celisse, *Segmentation of the mean of heteroscedastic data via cross-validation*, Arxiv preprint arXiv:0902.3977, 2009.

[9] Birgé, L. and Massart, P., *Gaussian model selection*, Journal of the European Mathematical Society, 3, 203–268, 2001.