UNCERTAINTY TREATMENT IN DISPERSION MODELLING OF ACCIDENTAL RELEASES

Vincent Dubourg[1], Patrick Armand[2], David Poulet[3], Florian Vendel[4], Sébastien Argence[3], Thierry Yalamas[1], Fabien Brocheton[3] and Perrine Volta[4]

[1] PHIMECA Engineering, F-63800 Cournon d'Auvergne, France
[2] CEA, DAM, DIF, F-91297 Arpajon, France
[3] NUMTECH, F-63175 Aubière, France
[4] SILLAGES Environnement, F-69130 Ecully, France

Abstract: This paper describes a probabilistic risk assessment framework for helping stake-holders making the best decisions right after an accidental release of pollutant. The imprecise release conditions (weather and source) are modelled as random variables. Uncertainty propagation through a Lagrangian dispersion model is then proposed so as to build probabilistic risk maps that would eventually help the rescue teams to settle a lanyard around the critical zone The proposed methodology is applied to a virtual phosphine accidental release in an actual constructed area.

Key words: *Lagrangian dispersion modelling, imprecise release conditions, probabilistic risk assessment, principal component analysis, Gaussian process predictors.*

INTRODUCTION

Since the most basic Gaussian plume dispersion model, physicists and engineers have significantly improved their models in order to better represent the dispersion of the flow of pollutants during either accidental or chronic releases through constructed areas. Indeed, the recent advances in the field of *computational fluid dynamics* (CFD) enable a better consideration of the effects of buildings on the atmospheric condition (the wind field) than the simple rugosity parameter that is typically used in the Gaussian plume model. Despite these unarguable improvements in the modelling of the dispersion phenomenon, numerical-simulation-based decision-making remains a hard task due to the numerous sources of uncertainty about the release. The latter argument is especially true for accidental releases which often lack accurate measurements of the meteorological conditions or of the emitted quantity of pollutant which are typically required to feed the dispersion model. In this paper, it is proposed to recourse to a probabilistic modelling for the uncertain release parameters and to propagate outcomes of these parameters through a Lagrangian dispersion model in order to assess the risk of exceeding a critical dose of air pollutant in the immediate surrounding of the release location. Of particular interest is the construction of animated time-varying risk maps for helping the stake-holders making the best decisions regarding evacuation, rescue teams settlement and prescription for medical treatment.

VIRTUAL TEST CASE: PHOSPHINE ACCIDENTAL RELEASE IN A CONSTRUCTED AREA

For illustration purposes, the proposed methodology is applied to a virtual accidental release of phosphine (chosen as an example) in an actual constructed area. The accidental release was imagined to occur at 06:35 am and to last for 5 minutes. The release setup is depicted in Figure 1. The yellow dot represents the source. The phosphine instant concentration is represented here 5 minutes after the beginning of the release at 1.5 meters above ground level for the two extreme wind directions considered in this study (215° and 234°).

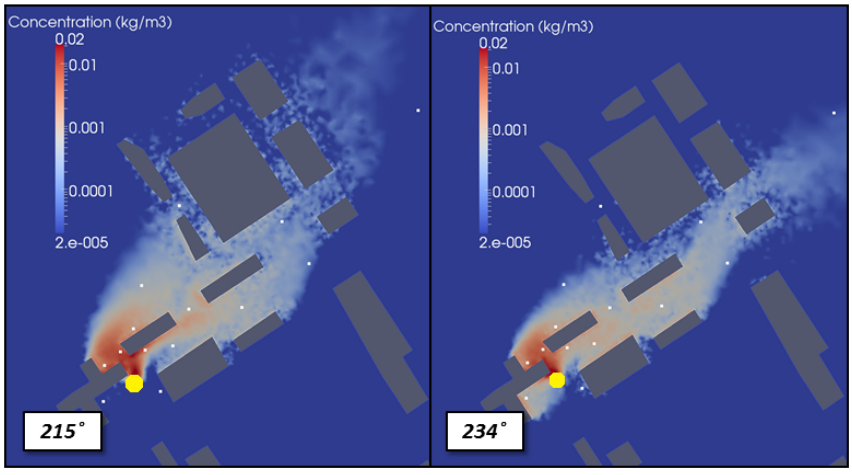


Figure 1. Virtual setup for an accidental release of phosphine in a constructed area.

The release parameters are assumed to be collected from imprecise data sources such as expert judgement and hence modelled as probability distributions. The uncertain parameters considered in this study are summarized in Table 1.

Table 1. Probabilistic model for the uncertain release conditions

Parameter	Probability distribution
Wind speed	Normal with mean 2 $m.s^{-1}$ and standard deviation 0.17 $m.s^{-1}$
Wind direction	Truncated normal with mean 225° and standard deviation 22.15°, over [215 ; 234°]
Cloud covering	Truncated normal with mean 6 octas and standard deviation 1 octa, over [1 octa; 9 octas]
Temperature	Uniform over [14°C ; 16°C]
Emitted quantity	Uniform over [70 $kg.s^{-1}$; 130 $kg.s^{-1}$]
Source height	Uniform over [1.75 m; 2.25 m]

DISPERSION MODELLING

This section briefly reviews the elements of the computational chain that has been developed in order to compute the dose of pollutant in the immediate surroundings of the source, during the minutes that follows the beginning of the release, for a given set of release conditions.

The safety atmospheric Lagrangian model

Accounting for constructed areas in dispersion modelling usually resorts to a computational fluid dynamics (CFD) model in order to assess the local atmospheric condition at every time step before advecting the pollutant particles using the conventional advection equations. This is known to be rather computationally demanding. The atmospheric air flow database methodology proposed and validated by Vendel *et al.* (2010) derives its numerical efficiency from the decoupling of the CFD-based calculation of the local wind field from the proper dispersion calculation.

The space- and time-dependent wind field is calculated using a CFD code on a chosen set of the meteorological parameters it depends on: the wind direction φ and the inverse Monin-Obukhov length $1/L_{MO}$. The calculated wind field is stored for later use at the time of the dispersion calculation. The set of meteorological parameters is actually selected as a regular grid in order to ease a linear interpolation of the wind field at any instant or point for a new set of meteorological parameters (see Vendel *et al.*, 2010 for details). The inverse Monin-Obukhov length is deduced from the cloud covering using a meteorological pre-processor. Once the wind field is calculated for the whole set of meteorological parameters, Vendel *et al.* (2010) have empirically shown that the computational time of the dispersion is reduced by a factor of 40 to 80 (compared to the usual coupled approach, illustrating the CPU cost of the CFD calculation) for similar results.

For the present study, the ANSYS FLUENT CFD model was run for 3 values of the inverse Monin-Obukhov length (-0,002; 0 and 0,002) and 5 wind directions (equally spaced between 215° and 235°), following the recommendations in Vendel *et al.* (2010). Eventually, the simulation of the 25 minutes of dispersion for a given set of release parameters requires about 10 minutes of CPU time using the sequential implementation of the *safety atmospheric Lagrangian model* (SLAM) developed at the Ecole Centrale de Lyon. SLAM was asked to yield concentration map snapshots every 30 seconds along the simulated time. The map is discretized over a two-dimensional grid containing 50 equally-spaced nodes, located at 1.5 meters above ground level which approximately covers the area depicted in Figure 1. The emitted pollutant is represented by 12 000 particles emitted at every second of the simulated release time. The concentration is estimated by counting the particles in square domains.

Calculation of the phosphine dose from the instant concentration

The effects of air pollutants on human health depend on the instant concentration level C, the time of exposure t, and the toxicity n. The *French institute for industrial environment and risks* (INERIS) estimated some threshold levels for different pollutant including the phosphine (PH_3, see Langlois, 2008). For risk assessment purposes, we define the pollutant dose from the instant concentration C according to the INERIS recommendations:

$$D(t) = \int_0^t C(\tau)^n \, d\tau \tag{1}$$

In the present study, this dose is calculated for the so-called *threshold of irreversible effects* (SEI for short, see Langlois, 2008). This means that we use a value of 0.53 for the toxicity exponent and the critical dose in the risk assessment is set to 20.10. In other words, the purpose of the present risk assessment study is to estimate the probability that the space- and time-varying dose D exceeds 20.10.

UNCERTAINTY PROPAGATION TECHNIQUES FOR RISK ASSESSMENT

This section presents the uncertainty propagation techniques that were used for estimating the probability of exceeding the critical threshold of pollutant on the spatiotemporal grid described above.

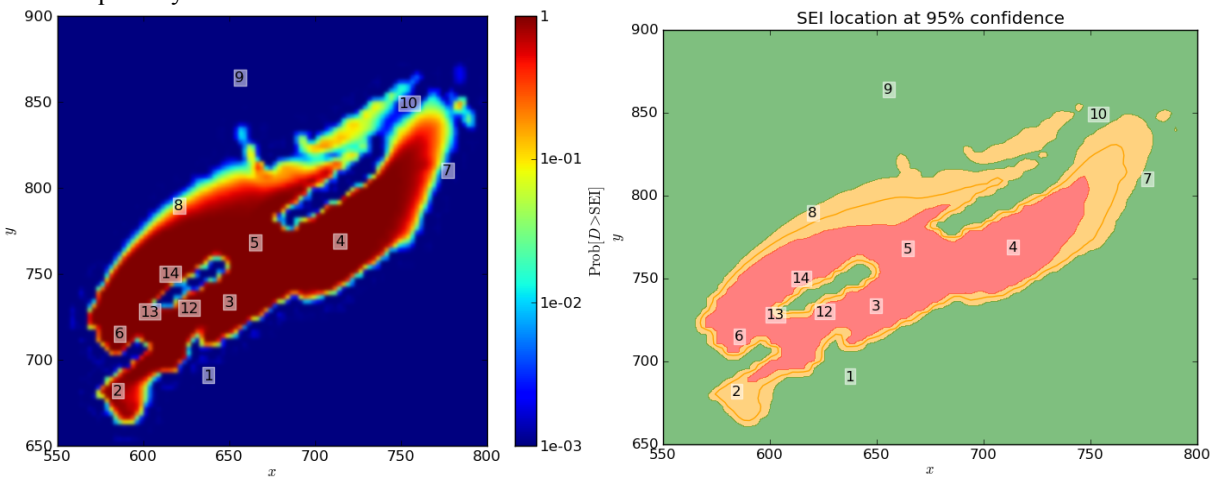
Brute force Monte Carlo sampling using high performance computing resources

The first intuitive strategy for estimating a probability is of course to recourse to Monte Carlo sampling of the release parameters through SLAM in order to collect an *N*-sample of the dose. Using this strategy, the probability estimate eventually reads:

$$\text{Prob}\left[D\left(\mathbf{x};\mathbf{p},t\right)\geq 20.10\right]\underset{N\to\infty}{\sim}\frac{1}{N}\sum_{i=1}^{N}\mathbb{I}\left[D\left(\mathbf{x}^{(i)};\mathbf{p},t\right)\geq 20.10\right] \tag{2}$$

where $\{D(\mathbf{x}^{(i)}; \mathbf{p}, t), i = 1, \ldots, N\}$ is the sample of dose at point $\mathbf{p}$ and time t calculated by SLAM for each release condition $\mathbf{x}^{(i)}$ drawn at random. This estimate is statistically consistent but it requires a sufficiently high sample size N, depending on the probabilities order of magnitude. As a rule of thumb, a confident estimate of a probability equal to 10^{-k} requires at least 10^{k+1} simulations. Here, probabilities lower than 10^{-3} will reasonably be considered to be zero, but the estimation of that upper bound would require about 10^4 SLAM runs.

Running SLAM 10^4 times sequentially on a single CPU would require more than 2 months. Such a delay is definitely not appropriate for accidental release. Hence, it is proposed to resort to a large cluster of PCs for accelerating the collection of the dose samples. In the present computational setup, 512 CPUs have been booked for the simulation in order to collect 10^3 dose samples within less than 20 minutes. Figure 2 represents the results of the probabilistic risk analysis estimated from these 10^3 samples. Panel (a) depicts the probability of exceeding the critical threshold while Panel (b) represents the 95% confidence interval on the location of that critical threshold. The green (resp. red) zone has 97.5% chances of being safe (resp. of not being safe). The orange zone is somewhat uncertain due to the imprecise release conditions. Despite the low sample size used for the estimation, these results constitute a sound reference for qualifying the ability of the second technique presented in the sequel to yield accurate results.



(a) Probability of exceeding SEI (b) SEI location and risk zones

Figure 2. Results of the probabilistic risk analysis obtained using brute force Monte Carlo sampling (snapshot at 06:40 am).

Surrogate-based Monte Carlo sampling using principal component analysis and Gaussian process predictors

In an attempt to reduce the number of SLAM runs, it is proposed to recourse to a surrogate model for predicting the dose using machine learning techniques. More specifically, a so-called *vector Gaussian process predictor* is built from a few samples of the dose, and this predictor is then used in order to calculate the dose for an arbitrarily high number of samples of the release conditions. Indeed the surrogate model is much faster to evaluate than the original model. This idea was already used for environmental risk engineering by Jia and Taflanidis (2012).

Surrogate models are statistics-based models that are built from a two-fold dataset (vector input $\mathbf{x}$ *vs.* scalar output y) in order to predict the output of another expensive-to-evaluate (physics-based) model. *Gaussian process* (GP) predictor (Santner *et al.*, 2003) is a specific prediction technique which uses a Bayesian formulation to come to this end. Santner *et al.* (2003) put a stationary GP prior on the original model:

$$Y\left(\mathbf{x}\right)=\mathbf{f}\left(\mathbf{x}\right)^{\mathrm{T}}\boldsymbol{\beta}+Z\left(\mathbf{x}\right) \tag{3}$$

where $\mathbf{f}(\mathbf{x})^{\mathrm{T}}\boldsymbol{\beta}$ is a linear regression model and Z is a stationary zero-mean GP with covariance function $C(\mathbf{x}, \mathbf{x}') = \sigma^2 R(\mathbf{x}, \mathbf{x}')$. Here, we resort to a constant for the regression model, meaning $\mathbf{f}(\mathbf{x}) = 1$ and $\boldsymbol{\beta} = \beta_0$, together with a squared exponential correlation function:

$$R(\mathbf{x}, \mathbf{x}') = \exp\left[\sum_{i=1}^{d}\left(\frac{x_i - x_i'}{\ell_i}\right)^2\right] \tag{4}$$

where $\boldsymbol{\ell}$ is a vector of correlation parameters that must be carefully selected for the sake of accuracy. Consequently, the observed model responses $\mathbf{Y} = (\mathcal{M}(\mathbf{x}^{(i)}),\ i = 1, \ldots, m)$ and some other unobserved model response $Y(\mathbf{x})$ are jointly distributed according to the following Gaussian distribution inherited from the GP prior:

$$\begin{pmatrix} \mathbf{Y} \\ Y(\mathbf{x}) \end{pmatrix} \sim \mathcal{N}_{m+1}\left(\begin{pmatrix} \mathbf{F}\boldsymbol{\beta} \\ \mathbf{f}(\mathbf{x})^{\mathrm{T}}\boldsymbol{\beta} \end{pmatrix}, \sigma^2\begin{bmatrix} \mathbf{R} & \mathbf{r}(\mathbf{x}) \\ \mathbf{r}(\mathbf{x})^{\mathrm{T}} & 1 \end{bmatrix}\right) \tag{5}$$

where $\mathbf{F} = [f_j(\mathbf{x}^{(i)}),\ i = 1, \ldots, m, j = 1, \ldots, p]$ and $\mathbf{R} = [R(\mathbf{x}^{(i)}),\ i, j = 1, \ldots, m]$ are the regression and correlation matrices, respectively; and $\mathbf{r}(\mathbf{x}) = (R(\mathbf{x}, \mathbf{x}^{(i)}),\ i = 1, \ldots, m)^{\mathrm{T}}$ is the cross-correlation vector. The sought predictor is then defined as the posterior distribution of some unobserved model response given the set of observed model responses (plus the GP prior parameters). Santner *et al.* (2003) showed that it is also a Gaussian random variable:

$$\hat{Y}(\mathbf{x}) = \left[Y(\mathbf{x})\middle|\mathbf{Y}; \ell, \sigma^2\right] \sim \mathcal{N}_1\left(\mu_{\hat{Y}}(\mathbf{x}), \sigma^2_{\hat{Y}}(\mathbf{x})\right) \tag{6}$$

with known mean and variance, see Santner *et al.* (2003) for the complete expressions. These expressions involve the generalized least-square solution for the regression weight vector $\boldsymbol{\beta}$. The covariance parameters are usually estimated from the data using either cross-validation (CV) or maximum likelihood estimation (MLE) techniques. The present implementation makes use of MLE (see Welch *et al.*, 1992).

Note that, in the prequel, Gaussian process predictors have been presented for emulating scalar-output models but the dispersion model output (the dose) is a function of space and time. This functional output has been converted to a vector output by discretization. Hence, one could construct a GP predictor for each component of the model output, although it would require a large amount of CPU time for both fitting and predicting. Starting from the premise that the output vector corresponds to a quantity that has been discretized over space and time, there must exist a strong correlation between its components. *Principal component analysis* (PCA) is then proposed so as to reduce the number of scalar GP predictors. The covariance matrix of the output vector is estimated from the observations in the dataset. It is then resolved for its largest eigenvalues $\lambda_1 \geq \lambda_2 \geq \ldots \geq \lambda_n \geq 0$, the stronger the correlation, the faster the decay of eigenvalues. Hence, it is reasonable to keep only the $r << n = \dim(\mathbf{y})$ eigensolutions that are associated with the largest eigenvalues. Based on this conclusion, the so-called Karhunen-Loève (KL) transform consists in mapping the original high-dimensional output vector to a lower dimensional vector in the subspace spanned by the significant eigensolutions. This *linear mapping* reads:

$$\mathbf{z} = \boldsymbol{\Lambda}_r^{-1/2}\boldsymbol{\Phi}_r^{\mathrm{T}}\mathbf{y} \tag{7}$$

where $\boldsymbol{\Lambda}_r = \mathrm{diag}([\lambda_i,\ \mathrm{i} = 1, \ldots, r])$ is the diagonal matrix of the r largest eigenvalues, and $\boldsymbol{\Phi}_r = [\boldsymbol{\varphi}_i,\ \mathrm{i} = 1, \ldots, r]$ is the orthogonal matrix of associated eigenvectors. The inverse KL transform reads:

$$\mathbf{y} = \boldsymbol{\Phi}_r\boldsymbol{\Lambda}_r^{1/2}\mathbf{z} \tag{8}$$

Eventually, only $r = \dim(\mathbf{z})$ scalar GP predictors are built in order to predict the model response in the space of principal components. The inverse KL transform is used to send the predictions back to its original dimension. Moreover, thanks to the linearity of the inverse KL transform, the Gaussian nature of the GP predictions is conserved and the final vector GP predictor reads:

$$\widehat{\mathbf{Y}}(x) = \mathcal{N}_n\left(\boldsymbol{\Phi}_r\boldsymbol{\Lambda}_r^{1/2}\boldsymbol{\mu}_{\mathbf{Z}}(\mathbf{x}), \boldsymbol{\Phi}_r\boldsymbol{\Lambda}_r^{1/2}\boldsymbol{\Sigma}_{\mathbf{ZZ}}(\mathbf{x})\left(\boldsymbol{\Phi}_r\boldsymbol{\Lambda}_r^{1/2}\right)^{\mathrm{T}}\right) \tag{9}$$

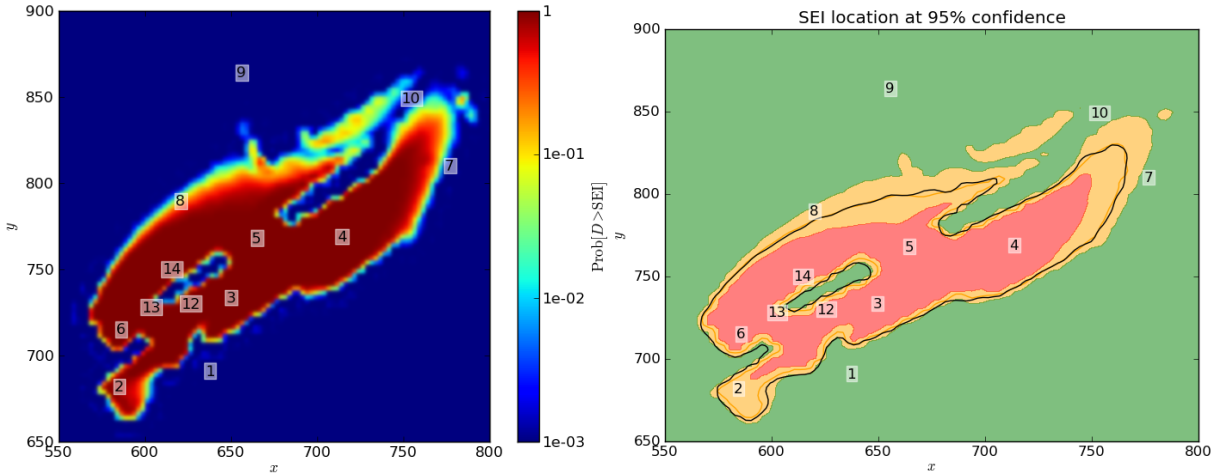
where $\boldsymbol{\mu}_{\mathbf{Z}}(\mathbf{x}) = (\mu_{Zi}(\mathbf{x}),\ i = 1, \ldots, r)^{\mathrm{T}}$ is the mean vector of the principal components GP predictors and $\boldsymbol{\Sigma}_{\mathbf{ZZ}}(\mathbf{x})$ is a diagonal matrix containing the associated variances of prediction.

The uncertainty in the vector GP predictor reflects the lack of information arising from the finite size dataset. It can be reduced by increasing the number of runs of the original code at a certain computational expense though. This new uncertainty is accounted for in the present risk analysis by using the following probability estimator (instead of the one given in Eq (2)):

$$\mathrm{Prob}\left[D(\mathbf{x}; \mathbf{p}, t) \geq 20.10\right] \underset{N\to\infty}{\sim} \frac{1}{N}\sum_{i=1}^{N} 1 - \Phi\left(\frac{20.10 - \mu_D\left(\mathbf{x}^{(i)}; \mathbf{p}, t\right)}{\sigma_D\left(\mathbf{x}^{(i)}; \mathbf{p}, t\right)}\right) \tag{10}$$

where Φ denotes the *cumulative distribution function* (CDF) of the standard Gaussian distribution, and μ_D (resp. σ_D) is the mean of the dose predicted by the vector GP predictor in Eq. (9) (resp. its standard deviation).

100 couples of release conditions and their corresponding dose have been selected in the sample used for the brute force Monte Carlo simulation. This selection had recourse to a *K*-means clustering technique based on the Euclidian distance in the space of the input parameters so as to cover the largest possible spectrum of release conditions (see the principle of space-filling *designs of experiments* (DOE) in the book by Santner *et al.*, 2003). A vector GP predictor has been built for each time step, hence PCA is performed on a vector of initial size 50 × 50 = 2 500. It is reduced to only 93 principal components fitted in about 10 minutes. *Leave-one-out cross-validation* (see e.g. Hastie *et al.*, 2009) is used in order to qualify the accuracy of the predictor. The leave-one-out estimate of the coefficient of determination (usually denoted by R^2) averaged on the whole map equals 0.45 for the 06:40 am snapshot. Then, the exceeding probability in Eq. (10) is estimated from 10^4 samples of the release conditions. The vector GP predictor requires less than 10 minutes for predicting the corresponding doses. The results obtained by means of this second approach are depicted in Figure 3. It can be seen that they are in reasonable agreement with the Monte Carlo reference results in Figure 2, in spite of the apparent poverty of the coefficient of determination. The increased uncertainty about the SEI location represented by the spread of the orange zone comes from the epistemic uncertainty introduced by the use of the vector GP predictor instead of the original dispersion model.



(a) Probability of exceeding SEI (b) SEI location and risk zones

Figure 3. Results of the probabilistic risk analysis obtained using a vector GP predictor as a surrogate for the dispersion model (snapshot at 06:40 am).

CONCLUSION

The proposed probabilistic risk assessment framework provides a sound reference for helping the stake-holders taking the safest possible decisions in the case of accidental releases by accounting for the whole uncertainty arising from *(i)* the imprecise release conditions *(ii)* the use of GP predictors instead of the original dispersion model for accelerating the decision-making process. Ultimately, provided the site CFD database is available, the risk maps can be built within a reasonably short time with respect to the urgency associated to an accident: about 20 minutes per SLAM run in the DOE (for the hundred used here and with a hundred CPUs, the overall time remains equal to the unit run time), 10 seconds for fitting the vector GP predictor and less than 30 seconds for predicting it 10 000 times for estimating the probability of interest at some chosen time step.

REFERENCES

Hastie, T.J., Tibshirani, R.J., and J.H. Friedman, 2009: *The elements of statistical learning. Second Edition*. Springer series in statistics. Springer.

Jia, G., and A.A. Taflanidis, 2012: Efficient hurricane risk assessment using kriging métamodèle. *11th ASCE joint specialty conference on probabilistic mechanics and structural reliability, Notre Dame, IN, USA*.

Langlois, C., 2008: Fiche INERIS DRC-08-94398-11851 A : Emissions accidentelles de substances chimiques dangereuses dans l'atmosphère. Seuils de toxicité aigüe pour la Phosphine. 3 pp.

Santner, T., Williams, B., and W. Notz, 2003: *The design and analysis of computer experiments*. Springer series in statistics. Springer.

Vendel, F., Lamaison, G., Soulhac, L., Donnat, L., Duclaux, O. and C. Puel, 2010: A new operational modelling approach for atmospheric dispersion in industrial complex area. *13th Int. Conf. on Harmo. within Atmos. Disp. Modell. for Regul. Purposes, Paris, France.* 266-270.

Welch, W., Buck, R., Sacks, J., Wynn, H., Mitchell, T. and M. Morris, 1992: Screening, predicting, and compute experiments. *Technometrics*. 34(1), 15-25.