



HAL
open science

Support vectors machines for the estimation of probability of failure: Multifidelity classifiers built from a posteriori discretization error estimators

Ludovic Pierre Jérôme Mell, Valentine Rey, Franck Schoefs

► To cite this version:

Ludovic Pierre Jérôme Mell, Valentine Rey, Franck Schoefs. Support vectors machines for the estimation of probability of failure: Multifidelity classifiers built from a posteriori discretization error estimators. Structural Safety, 2023, 102, pp.102321. <10.1016/j.strusafe.2023.102321>. <hal-04095640>

HAL Id: hal-04095640

<https://hal.science/hal-04095640v1>

Submitted on 12 May 2023

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



HAL Authorization

STRUCS 102321

Article reference: STRUCS STRUCS-D-22-00130

Support vectors machines for the estimation of probability of failure:
multifidelity classifiers built from a posteriori discretization error
estimators

Ludovic Mell¹, Valentine Rey^{1,*}, Franck Schoefs¹

¹ Nantes Université, École Centrale Nantes, CNRS, GeM, UMR 6183, F-44000 Nantes, France

* corresponding author valentine.rey@univ-nantes.fr

Abstract

This article proposes two algorithms to compute the probability of failure using Support Vector Machine (SVM) classifiers and Monte Carlo estimator in the context of structural mechanics. The observations used to build the classifiers are obtained from calls to a finite element solver which introduces discretization error. By exploiting guaranteed discretization error estimators, the proposed methodology aim at computing an estimation of the probability of failure not polluted by this discretization error. The first algorithm builds two classifiers in parallel to separate the guaranteed fail population, the guaranteed safe population and the uncertain population. It enables to compute an upper bound and a lower bound of the exact probability of failure. The second algorithm only uses observations whose status (fail or safe) is guaranteed by the error estimators. It results in multi-fidelity SVM-based meta-models as observations computed on different mesh sizes can be used. Those two algorithms are illustrated on two-dimensional mechanical examples for different Monte Carlo populations.

Keywords: Reliability, Probability of failure, Support Vector Machines, Finite Element Method, Discretization Error

1. Introduction

2 Many industrial structures are used in an uncertain environment. For example, structures
3 may be subjected to wind loads or wave loads that are not deterministic but random. Moreover,
4 as the perfect knowledge of the structure itself is impossible, its geometry or its material prop-
5 erties can also be a source of uncertainty and are sometimes modeled as random variables. In

6 this context, the design of the structure requires taking into account those uncertainties, which
7 is usually done by reliability analysis.

8 Reliability analysis consists in studying the ability of a structure to accomplish its function
9 throughout its lifetime. A failure scenario is modeled by a performance function G , also called
10 limit state function, which is the difference between a resistance and a solicitation. A negative
11 limit state function corresponds to failure and a strictly positive state function corresponds
12 to safety. Generally, the uncertainties on the mechanical structure (load, geometry, material
13 properties) are modeled as random variables following identified distributions. As a consequence,
14 reliability analysis consists in computing sensitivity factors, reliability indexes or a probability
15 of failure, which is the probability that the performance function is negative. In this article, we
16 focus on the estimation of the probability of failure.

17 Usually, for industrial structures, no explicit form of G with respect to the random variables
18 is available. Therefore, even if the joint probability of the random variables is known, it is not
19 possible to directly compute the probability of failure. However, a mechanical model of the
20 structure is postulated and the arising equations can be solved, usually thanks to discretization
21 techniques, such as the finite element method (FEM) [1]. FEM is nowadays largely used in
22 commercial softwares and enables simulating and predicting the response of a structure. As a
23 consequence, the probability of failure can be estimated using Monte Carlo estimators [2] in a
24 non-intrusive way. In spite of their simplicity, these methods suffer a poor convergence rate: a
25 precise estimation of the probability of failure (usually around 10^{-4}) requires the computation of
26 $G(x)$, that is to say a call to the FEM code, for every point x of an extremely large Monte Carlo
27 population. As a consequence, variance reduction techniques, such as importance sampling [3],
28 subset sampling [4], Multi-Level Monte Carlo [5], adjusted control variates techniques [6] have
29 been proposed. Another way to reduce computational costs consists in building a meta-model
30 \hat{G} that would be a satisfying cheap approximation of G . Kriging-based meta-models [7] are very
31 popular in reliability analysis, but physical response surfaces [8], polynomial responses surfaces,
32 radial basis functions or neuronal networks, see [9], can also be considered.

33 As the definition of the probability of failure only relies on the sign of G , classifiers are
34 interesting candidates to build a meta-model. Among them, support vector machines (SVM)
35 [10] are widely used in the context of reliability analysis, see [11, 12, 13, 14, 15]. The classifier
36 enables separating the Monte Carlo population into two sub-populations: the failure population
37 and the safety population. A learning criterion is available to improve the quality of the classifier
38 from calls to the FEM code. The resulting algorithms are competitive and require comparable

39 number of observations comparing to standard kriging methods.

40 It is well-known that the result of FEM computations depends on the discretization [16].
41 A too coarse mesh might lead to wrong displacements and therefore introduces an error, called
42 discretization error, in the computations of all the mechanical outputs (average stress, ...) leading
43 to a misclassification. It is crucial to take into account this discretization error in the reliability
44 analysis [17, 18]. To this day, only few works address the question of the consequence of the
45 discretization error on the estimation of the probability of failure. It was done with response
46 surface methodology [19], with FORM in [20] or in the context of kriging with co-kriging [21].
47 Discretization error estimators are available in the literature [16] and enable computing the error
48 on a quantity of interest of the mechanical problem. Those estimators were exploited in the
49 context of reliability within FORM [22] and kriging [23]. However, those approaches are limited
50 to certain type of meta-models and do not include SVM.

51 In this paper, we propose to exploit a posteriori error estimators to control the influence of
52 the discretization error during the construction of an adaptive SVM-based classifier. We give two
53 algorithms that compute the probability of failure. The first one builds two classifiers: the first
54 classifier separates the guaranteed safe domain from its complementary and the second classifier
55 separates the guaranteed fail domain from its complementary. Therefore, we can compute upper
56 and lower bound of the probability of failure. The second algorithm only uses learning points
57 for which the sign is not polluted by the discretization error and aims at using coarse meshes far
58 from the limit state and fine meshes only close to the limit state. It results in a multi-fidelity
59 meta-model because its construction relies on computations done on two different meshes. These
60 two algorithms are illustrated on two mechanical examples with 2 random variables: a synthetic
61 example for which the exact solution is known and a cracked plate for which the failure scenario is
62 the propagation of the crack. In Section 2, we formulate the mechanical problem, define the limit
63 state function, the probability of failure and derive bounds on values of the limit state function.
64 In Section 3, we give the principles of SVM and briefly describe its use for the estimation of the
65 probability of failure. The two new algorithms are presented in Section 4. Numerical simulations
66 are done in Section 5. The last section concludes the paper.

67 2. Description of the mechanical reliability problem

68 In this section, we first define the continuous mechanical problem and its discretization. Then,
69 we show how the use of a *a posteriori* error estimator based on the constitutive relation enables to
70 obtain discretization error bounds on the exact value of the limit state function $G_{ex}(x)$. For the

71 sake of simplicity, these derivations are done in a deterministic framework. In the last subsection,
 72 we introduce uncertainties in the form of random variables and explicit the link between the limit
 73 state function and the probability of failure.

74 2.1. Continuous problem

75 Let us consider a body Ω occupying the physical space \mathbb{R}^3 that is subjected to a body force
 76 \underline{f}_{vol} , to an imposed displacement \underline{u}_d on $\partial_u\Omega$ whose measure is not null and to external forces
 77 \underline{F} on $\partial_F\Omega$. Note that $\partial_F\Omega \cup \partial_u\Omega = \partial\Omega$ and that $\text{meas}(\partial_F\Omega \cap \partial_u\Omega) = 0$. We consider that this
 78 body undergoes small perturbations and that there is no inertia effect so that the evolution is
 79 quasi-static. We also assume that the material behaviour of this solid Ω is linear elastic so that
 80 it can be characterized by the Hooke tensor \mathbb{H} . We note \underline{u} the displacement of the body and $\underline{\sigma}$
 81 the Cauchy stress tensor.

82 Two affine subspaces and a positive form are introduced:

- 83 • The affine subspace of kinematically admissible fields (KA-fields)

$$\text{CA} = \left\{ \underline{u} \in (\mathbb{H}^1(\Omega))^d, \underline{u} = \underline{u}_d \text{ on } \partial_u\Omega \right\} \quad (1)$$

84 and we note CA^0 the associated vectorial space.

- Affine subspace of statically admissible fields (SA-fields)

$$\text{SA} = \left\{ \underline{\tau} \in (\mathbb{L}^2(\Omega))_{\text{sym}}^{d \times d}; \forall \underline{v} \in \text{CA}^0, \int_{\Omega} \underline{\tau} : \underline{\varepsilon}(\underline{v}) d\Omega = \int_{\Omega} \underline{f}_{vol} \cdot \underline{v} d\Omega + \int_{\partial_F\Omega} \underline{F} \cdot \underline{v} dS \right\} \quad (2)$$

- 85 • Error in constitutive equation

$$e_{CR\Omega}(\underline{u}, \underline{\sigma}) = \|\underline{\sigma} - \mathbb{H} : \underline{\varepsilon}(\underline{u})\|_{\mathbb{H}^{-1}, \Omega} \quad (3)$$

86 where $\|\underline{x}\|_{\mathbb{H}^{-1}, \Omega} = \sqrt{\int_{\Omega} (\underline{x} : \mathbb{H}^{-1} : \underline{x}) d\Omega}$

87 The continuous problems reads:

$$\left\{ \begin{array}{l} \text{Find a displacement field } \underline{u} \text{ and a stress field } \underline{\sigma} \text{ such that} \\ \underline{u} = \underline{u}_d \text{ on } \partial_u\Omega \text{ and } \underline{\varepsilon}(\underline{u}) = \frac{1}{2}(\underline{\text{grad}}(\underline{u}) + \underline{\text{grad}}^T(\underline{u})) \text{ on } \Omega \\ \underline{\text{div}}(\underline{\sigma}) + \underline{f}_{vol} = \underline{0} \text{ on } \Omega \text{ and } \underline{\sigma} \underline{n} = \underline{F} \text{ on } \partial_F\Omega \\ \underline{\sigma} = \mathbb{H} : \underline{\varepsilon}(\underline{u}) \text{ on } \Omega \end{array} \right. \quad (4)$$

88 The following formulation is equivalent to the problem 4.

89 Find $(\underline{u}_{ex}, \underline{\sigma}_{ex}) \in \text{CA} \times \text{SA}$ such that $e_{CR\Omega}(\underline{u}_{ex}, \underline{\sigma}_{ex}) = 0$

90 Under the hypothesis presented at the beginning of this section, the exact solution $(\underline{u}_{ex}, \underline{\sigma}_{ex})$
 91 exists and is unique.

92 In the context of reliability, one defines a limit state function or performance function G that
 93 is usually written as a difference between a resistance R and a solicitation S : $G = R - S$. In this
 94 paper, we consider that the solicitation is a linear form of the displacement

$$S = \tilde{L}(\underline{u}) \quad (5)$$

95 If $G \leq 0$, the structure fails. If $G > 0$, the structure is considered safe.

96 Unfortunately, the analytical form of the solution \underline{u}_{ex} is unavailable so $S(\underline{u}_{ex})$ and $G_{ex} = R -$
 97 $S(\underline{u}_{ex})$ are generally unknown. Therefore, this problem is usually solved thanks to discretization
 98 technique. In this paper, we consider that the finite element method (FEM) is used.

99 2.2. Discrete problem

100 Let define Ω_H a tessellation of Ω . Continuous shape functions are associated to this tessel-
 101 lation. The FEM relies on the approximation of the subspace of kinematically admissible fields
 102 by a subspace of finite dimension:

$$\text{CA}_H = \left\{ \underline{u} \in (\text{H}^1(\Omega_H))^d, \underline{u} = \underline{u}_d \text{ on } \partial_u \Omega_H \right\} \quad (6)$$

103 We note CA_H^0 the associated vectorial space.

104 Therefore, the discretized problem reads:

$$\begin{aligned} &\text{Find a displacement field } \underline{u}_H \in \text{CA}_H \text{ such that } \forall \underline{v}_H \in \text{CA}_H^0 \\ &\int_{\Omega_H} \underline{\underline{\varepsilon}}(\underline{u}_H) : \mathbb{H} : \underline{\underline{\varepsilon}}(\underline{v}_H) d\Omega = \int_{\Omega_H} \underline{f}_{vol} \cdot \underline{v}_H d\Omega + \int_{\partial_F \Omega_H} \underline{F} \cdot \underline{v}_H dS \end{aligned} \quad (7)$$

105 Once \underline{u}_H is obtained, an approximated stress field can be computed thanks to the constitutive
 106 relation:

$$\underline{\sigma}_H = \mathbb{H} : \underline{\underline{\varepsilon}}(\underline{u}_H) \quad (8)$$

107 However, the finite element solution \underline{u}_H almost never coincides with the exact solution \underline{u}_{ex} , so
 108 $R - S(\underline{u}_{ex}) \neq R - S(\underline{u}_H)$ which is the reason why the FEM introduces a discretization error that
 109 is propagated in the reliability analysis. Note that choosing *a priori* an optimal mesh for the

110 reliability study is impossible as the error on S may non linearly depend on the random variables
 111 and that the conception point is usually unknown [17], even more if the resistance R is random.

112 In the next subsection, we define the discretization error and present a posteriori error esti-
 113 mators.

114 2.3. Estimation of the discretization error

115 2.3.1. Generalities

116 One can introduce the displacement discretization error defined as $e_{discr} = \underline{u}_{ex} - \underline{u}_H$ and its
 117 energetic norm $\|e_{discr}\|_{\Omega} := \|\underline{\underline{\varepsilon}}(\underline{u}_{ex} - \underline{u}_H)\|_{\mathbb{H}^{-1},\Omega}$. The Pragger-Synge theorem [24] reads:

$$\begin{aligned} \forall(\underline{\hat{u}}, \underline{\hat{\sigma}}) \in \text{CA}(\Omega) \times \text{SA}(\Omega), \\ \|\underline{\underline{\varepsilon}}(\underline{u}_{ex}) - \underline{\underline{\varepsilon}}(\underline{\hat{u}})\|_{\mathbb{H},\Omega}^2 + \|\underline{\underline{\sigma}}_{ex} - \underline{\underline{\hat{\sigma}}}\|_{\mathbb{H}^{-1},\Omega}^2 = e_{CR\Omega}^2(\underline{\hat{u}}, \underline{\hat{\sigma}}) \end{aligned} \quad (9)$$

118 This equality is the key ingredient of the discretization error estimator based on the consti-
 119 tutive relation. Indeed, as the finite element solution $\underline{u}_H \in \text{CA}_H$, one can choose $\underline{\hat{u}} = \underline{u}_H$ and
 120 then we can write:

$$e_{discr} := \|e_{discr}\|_{\Omega} \leq e_{CR\Omega}(\underline{u}_H, \underline{\hat{\sigma}}) \quad (10)$$

121 This inequality proves that $e_{CR\Omega}(\underline{u}_H, \underline{\hat{\sigma}})$ is a guaranteed *a posteriori* error estimator. To compute
 122 this estimator, it is required to build a statically admissible stress field $\underline{\hat{\sigma}}$. This task may be
 123 complex but several techniques are available in the litterature (see [25], [26], [27] and [28]).

124 The energetic norm of the discretization error is often difficult to use in an industrial context.
 125 Indeed, engineers are often more interested in the error done for a specific quantity of interest
 126 such as a stress component in a region of the structure. The next paragraph shows how this
 127 error estimator can be used to obtain discretization error bounds on a quantity of interest that
 128 is the solicitation S_H .

129 2.3.2. Definition of the linear quantity of interest and the adjoint problem

130 We propose to write G as:

$$G = R - \tilde{L}(\underline{u}) = R - \int_{\Omega} (\underline{\underline{\sigma}}_{\Sigma} : \underline{\underline{\varepsilon}}(\underline{u}) + \underline{f}_{\Sigma} \underline{u}) d\Omega \quad (11)$$

131 In this equation $\underline{\underline{\sigma}}_{\Sigma}$ and \underline{f}_{Σ} are called extractors and depend on the nature of the quantity
 132 of interest which is the solicitation S in a reliability analysis. For example, if the quantity of

133 interest is the displacement at the point \underline{P} on the \underline{e}_1 direction, then the extractors are $\underline{\underline{\sigma}}_\Sigma = \underline{\underline{0}}$
 134 and $\underline{f}_\Sigma = \delta(\underline{P})\underline{e}_1$.

135 A second mechanical problem, also called adjoint problem, is defined:

$$\text{Find } \left(\underline{\tilde{u}}_{ex}, \underline{\tilde{\underline{\sigma}}}_{ex} \right) \in \text{CA}^0(\Omega) \times \widetilde{\text{SA}}(\Omega) \text{ such that } e_{CR\Omega}(\underline{\tilde{u}}_{ex}, \underline{\tilde{\underline{\sigma}}}_{ex}) = 0 \quad (12)$$

136 where

$$\widetilde{\text{SA}}(\Omega) = \left\{ \underline{\underline{\tau}} \in (\text{L}^2(\Omega))_{\text{sym}}^{d \times d}; \forall \underline{v} \in \text{CA}^0(\Omega), \int_{\Omega} \underline{\underline{\tau}} : \underline{\underline{\varepsilon}}(\underline{v}) d\Omega = \tilde{L}(\underline{v}) \right\} \quad (13)$$

137 The solution to this problem exists and is unique. Since the exact solution is not known, this
 138 problem can be solved with the finite element method. By introducing a tessellation \tilde{H} , we then
 139 define the vectorial subspace :

$$\text{CA}_{\tilde{H}}^0 = \left\{ \underline{u} \in (\text{H}^1(\Omega_{\tilde{H}}))^d, \underline{u} = 0 \text{ on } \partial_u \Omega_H \right\} \quad (14)$$

140 The discretized adjoint problem reads:

$$\begin{aligned} &\text{Find a displacement field } \underline{\tilde{u}}_{\tilde{H}} \in \text{CA}_{\tilde{H}}^0 \text{ such that } \forall \underline{v}_{\tilde{H}} \in \text{CA}_{\tilde{H}}^0 \\ &\int_{\Omega_{\tilde{H}}} \underline{\underline{\varepsilon}}(\underline{\tilde{u}}_{\tilde{H}}) : \mathbb{H} : \underline{\underline{\varepsilon}}(\underline{v}_{\tilde{H}}) d\Omega = \tilde{L}(\underline{v}_{\tilde{H}}) \end{aligned} \quad (15)$$

141 One can observe that this problem is similar to the reference finite element problem. The only
 142 difference is that the linear form \tilde{L} has replaced the mechanical forces \underline{f}_{vol} and \underline{F} . Therefore, if
 143 one choses to use the same tessellation for the reference and the adjoint problems ($\tilde{H} = H$) then
 144 both problems can be solved at the same time since they share the same stiffness matrix which
 145 enables multiple right-hand side solving. The use of the FEM for the adjoint problem introduces
 146 a discretization error that can be estimated in the same way as the reference mechanical problem:

$$\tilde{e}_{discr} = \|\underline{\tilde{u}}_{ex} - \underline{\tilde{u}}_{\tilde{H}}\|_{\Omega} \leq e_{CR\Omega}(\underline{\tilde{u}}_{\tilde{H}}, \hat{\underline{\underline{\sigma}}}_{\tilde{H}}) \quad (16)$$

147 where $\hat{\underline{\underline{\sigma}}}_{\tilde{H}}$ is a statically admissible stress field for the adjoint problem.

148 2.3.3. Error estimation on a quantity of interest

149 In [29], the following inequality is demonstrated:

$$|\tilde{L}(\underline{u}_{ex}) - \tilde{L}(\underline{u}_H) - I_{H\tilde{H}}| \leq \frac{1}{2} e_{CR\Omega}(\underline{u}_H, \hat{\underline{\underline{\sigma}}}_{\tilde{H}}) e_{CR\Omega}(\underline{\tilde{u}}_{\tilde{H}}, \hat{\underline{\underline{\sigma}}}_{\tilde{H}}) \quad (17)$$

150 where

$$I_{H\tilde{H}} = \int_{\Omega} \frac{1}{2} (\hat{\underline{\underline{\sigma}}}_{\tilde{H}} + \mathbb{H} : \underline{\underline{\varepsilon}}(\underline{\tilde{u}}_{\tilde{H}})) : \mathbb{H}^{-1} : (\hat{\underline{\underline{\sigma}}}_{\tilde{H}} - \mathbb{H} : \underline{\underline{\varepsilon}}(\underline{u}_H)) d\Omega \quad (18)$$

151 and where $\hat{\underline{\sigma}}_{\tilde{H}} \in \tilde{\text{SA}}_{\tilde{H}}(\Omega)$. Therefore, by defining

$$G^m = R - \tilde{L}(\underline{u}_H) + I_H \tilde{H} \quad (19)$$

152 we can derive bounds on the exact value of $G_{ex} = R - \tilde{L}(\underline{u}_{ex})$

$$G^- \leq G_{ex} \leq G^+ \quad (20)$$

153 where

$$G^- := G^m - \frac{1}{2} e_{CR\Omega}(\underline{u}_H, \hat{\underline{\sigma}}_{\tilde{H}}) e_{CR\Omega}(\tilde{\underline{u}}_{\tilde{H}}, \hat{\underline{\sigma}}_{\tilde{H}}) \quad (21)$$

154 and

$$G^+ := G^m + \frac{1}{2} e_{CR\Omega}(\underline{u}_H, \hat{\underline{\sigma}}_{\tilde{H}}) e_{CR\Omega}(\tilde{\underline{u}}_{\tilde{H}}, \hat{\underline{\sigma}}_{\tilde{H}}) \quad (22)$$

155 Note that G_H may be outside of $[G^-; G^+]$. As a consequence, G^m is usually a better esti-
156 mation of G_{ex} than $R - \tilde{L}(\underline{u}_H)$.

157 2.3.4. Probability of failure

158 Let us consider that the uncertainties can be modeled by a vector random variable $X : \zeta \in$
159 $Z \rightarrow X(\zeta) = \underline{x} \in \mathbb{R}^p$ where Z is the set of possible outcomes and $\underline{x} \in \mathbb{R}^p$ a realization of this
160 random variable. The random variables can be the imposed forces or imposed displacements, a
161 parameter of the geometry, a material property, ... As a consequence, the displacement \underline{u} and
162 the quantity of interest $\tilde{L}(\underline{u})$ are also random variables.

163 Let us note p the joint distribution of X . The exact probability of failure is:

$$P_{f,ex} = \int_{D_{ex,f}} p(\underline{x}) d\underline{x} \quad (23)$$

164 where $D_{ex,f} = \{\underline{x} \in \mathbb{R}^p | G_{ex}(\underline{x}) \leq 0\}$ is the exact failure domain. However, the exact limit
165 state function G_{ex} being usually unknown, the computation of the exact probability of failure is
166 impossible and G_{ex} is approximated by G_H to compute the probability of failure

$$P_{f,H} = \int_{D_{H,f}} p(\underline{x}) d\underline{x} \quad (24)$$

167 where $D_{H,f} = \{\underline{x} \in \mathbb{R}^p | G_H(\underline{x}) \leq 0\}$ is the finite element failure domain. Of course, due to the
168 discretization error, $G_{ex} \neq G_H$ so $D_{ex,f} \neq D_{H,f}$ and therefore $P_{f,H} \neq P_{f,ex}$. Moreover, G_H
169 is not known explicitly but implicitly, which makes the analytical evaluation of the probability
170 $P_{f,H}$ impossible. Additionnaly, there is no warranty that $P_{f,H}$ exceeds or not $P_{f,ex}$, so that

171 the approximation may be conservative or not. Finally, this sum is usually estimated using the
 172 Monte Carlo method. Even though the Monte Carlo estimation introduces an approximation
 173 error, it can be controlled by ensuring that the Monte Carlo population is large enough. The
 174 coefficient of variation (COV) of the probability is a widely used indicator of this approximation.

175 Note that the computation of the integral does not require knowing the value of G_H on
 176 \mathbb{R}^p but only the sign of G_H through the definition of the failure domain. This is the reason
 177 why classification methods are relevant tools to estimate the probability of failure. Once the
 178 classifier is built, the classification of the Monte Carlo population and then the computation of
 179 the probability of failure are cheap in terms of computational cost even if the population is large.
 180 In the next subsection, the support vector machine classification is briefly presented.

181 3. Support Vector Machine (SVM) for classification

182 In this section, we briefly describe the basics on both linear SVM and non linear SVM for
 183 the classification of a point $(\mathbf{x}, y) \in \mathbb{R}^p \times \{-1; 1\}$ from n data points, also called observations
 184 $(\mathbf{x}_i, y_i)_{i=1..n}$. The objective of this method is to build a classifier $D : \mathbb{R}^p \rightarrow \{-1; 1\}$. The
 185 interested reader can complete this short section on the basics of SVM classification with the
 186 following reference [10].

187 3.1. Linear SVM

188 If the data are linearly separable, the classifier D may be built from the linear function f
 189 defined as:

$$f(\mathbf{x}) = \mathbf{v}^T \mathbf{x} + a \text{ where } a \in \mathbb{R} \text{ and } \mathbf{v} \in \mathbb{R}^p \quad (25)$$

190 where $\mathbf{v}^T \mathbf{x}$ is the scalar product between \mathbf{v} and \mathbf{x} . Therefore the classifier D reads:

$$D(\mathbf{x}) = \text{sign}(f(\mathbf{x})) \quad (26)$$

191 where

$$\text{sign}(\square) = \begin{cases} 1 & \text{if } \square > 0 \\ -1 & \text{otherwise} \end{cases} \quad (27)$$

192 The hyperplane Δ is defined as $\Delta = \{\mathbf{x} \in \mathbb{R}^p \text{ such that } f(\mathbf{x}) = 0\}$ and the margin m is the
 193 minimal distance between the data and the hyperplane Δ :

$$m = \min_{i=1..n} \left(\frac{|\mathbf{v}^T \mathbf{x}_i + a|}{\|\mathbf{v}\|} \right) \quad (28)$$

194 The parameters \mathbf{v} and a of the function f are sought such that they maximize the margin m . To
 195 obtain a unique solution, this optimization problem is rewritten with new variables $\mathbf{w} = \frac{\mathbf{v}}{m\|\mathbf{v}\|}$
 196 and $b = \frac{a}{m\|\mathbf{v}\|}$ and reads :

$$\text{Find } \mathbf{w} \text{ and } b \text{ such that } \frac{1}{2}\|\mathbf{w}\|^2 \text{ is minimum and } y_i(\mathbf{w}^T \mathbf{x}_i + b) \geq 1 \quad \forall i = 1..n \quad (29)$$

197 This problem is called the primal problem as the unknowns $w = \frac{v}{m\|v\|}$ and $b = \frac{a}{m\|v\|}$ can be
 198 directly connected to the function $\tilde{f} = \mathbf{w}^T \mathbf{x} + b$ written with the new variables and defining the
 199 classifier D . Indeed, $\forall \mathbf{x} \in \mathbb{R}^p$, $\text{sign}(f(\mathbf{x})) = \text{sign}(\tilde{f}(\mathbf{x}))$.

200 It is possible to obtain a dual formulation of this problem by writing the Lagrangian \mathcal{L} of
 201 problem (29). The Lagrangian reads:

$$\mathcal{L}(\mathbf{w}, b, \alpha) = \frac{1}{2}\|\mathbf{w}\|^2 - \sum_{i=1}^n \alpha_i (y_i(\mathbf{w}^T \mathbf{x}_i + b) - 1) \quad (30)$$

202 with α_i the Lagrange multipliers associated to the inequality constraint. The saddle point opti-
 203 mality conditions are:

$$\left\{ \begin{array}{l} \mathbf{w} - \sum_{i=1}^n \alpha_i (y_i \mathbf{x}_i) = \mathbf{0} \\ \sum_{i=1}^n \alpha_i y_i = 0 \\ \forall i \in [1; n], \alpha_i \geq 0 \\ \forall i \in [1; n], (y_i(\mathbf{w}^T \mathbf{x}_i + b) - 1) \geq 0 \\ \forall i \in [1; n], \alpha_i (y_i(\mathbf{w}^T \mathbf{x}_i + b) - 1) = 0 \end{array} \right. \quad (31)$$

204 Let $\mathcal{A} = \{i \in [1; n] \mid (y_i(\mathbf{w}^T \mathbf{x}_i + b) - 1) = 0\}$ denote the set of active constraints. Then, the
 205 stationarity of the Lagrangian reads:

$$\mathbf{w} = \sum_{i \in \mathcal{A}} \alpha_i y_i \mathbf{x}_i \quad (32)$$

206 The vector \mathbf{w} is a linear combination of the observations associated to the active constraints:
 207 those observations are called support vectors. The other data do not affect the definition of the
 208 hyperplane. One can replace the primal variables b and \mathbf{w} in the primal formulation (29) to
 209 obtain the dual formulation:

$$\begin{aligned} \text{Find } \alpha_i \text{ for } i \in [1; n] \text{ such that } \frac{1}{2} \sum_{i=1}^n \sum_{j=1}^n \alpha_i \alpha_j y_i y_j \mathbf{x}_i^T \mathbf{x}_j - \sum_{i=1}^n \alpha_i \text{ is minimum and} \\ \sum_{i=1}^n \alpha_i y_i = 0 \text{ and } \alpha_i \geq 0 \quad \forall i = 1..n \end{aligned} \quad (33)$$

210 Both primal and dual problems can be solved with standard optimization solvers such as quadratic
 211 programming for instance.

212 3.2. Non-linear SVM

213 Unfortunately, in general cases, the optimal classifier is not linear. To overcome this difficulty,
 214 non-linear kernels may be introduced. The kernels are used to replace the scalar product $\mathbf{x}_i^T \mathbf{x}_j$
 215 by a measure of the influence of \mathbf{x}_i on \mathbf{x}_j . This influence is noted $\kappa(\mathbf{x}_i, \mathbf{x}_j)$ and can be interpreted
 216 as a correlation. Therefore, only positive kernels are allowed. One can cite polynomial kernel,
 217 Gaussian kernel, exponential kernel. In this article, we use the Gaussian kernel:

$$\kappa(\mathbf{x}_i, \mathbf{x}_j) = \exp\left(-\frac{\|\mathbf{x}_i - \mathbf{x}_j\|_2^2}{2\sigma^2}\right) \quad (34)$$

218 where σ is an hyperparameter. In this paper, we do not study the influence of the choice of the
 219 kernel. However, methods to build an optimal kernel from the observations exist [30]. The choice
 220 of the hyperparameter σ is of prime importance. In this article, we perform cross-validation as
 221 available in Mathworks. The dual formulation of the non-linear SVM reads:

$$\begin{aligned} \text{Find } \alpha_i \text{ for } i \in [1; n] \text{ such that } & \frac{1}{2} \sum_{i=1}^n \sum_{j=1}^n \alpha_i \alpha_j y_i y_j \kappa(\mathbf{x}_i^T \mathbf{x}_j) - \sum_{i=1}^n \alpha_i \text{ is minimum and} \\ & \sum_{i=1}^n \alpha_i y_i = 0 \text{ and } 0 \leq \alpha_i \leq C \forall i = 1..n \end{aligned} \quad (35)$$

222 One can observe the introduction of the parameter C which is a penalty parameter. Choosing
 223 a finite value of C authorizes misclassification. In this paper, we choose to increase C to infinity
 224 which means that the values α_i are not bounded and therefore that misclassification is not
 225 allowed. Once again, this optimization problem can be solved with standard solvers.

226 3.3. SVM classification for estimation of the probability of failure

227 Classification with SVM has already been used to estimate the probability of failure [31, 12, 13]
 228 in the context of reliability analysis. In [12], the authors build a SVM classifier to separate
 229 the failure points from the safety points in the Monte Carlo Population U of size n_{MC} . From
 230 observations on a design of experiment *DOE*, a SVM classifier is built. Then, a learning criterion
 231 and two learning functions are proposed to improve the classifier adding new observations. Let ξ
 232 be the ratio between the number of points of U located inside the margin of the classifier over the
 233 size of U . Let k be the number of the learning iteration. In order to have a more stable criterion,
 234 the authors of [12] propose to fit the points of ξ using an exponential curve and then define

235 $\hat{\xi}(k) = A \exp(Bk)$. If $\xi > \eta_1$ or $\hat{\xi} > \eta_1$ or $\left| \frac{d\hat{\xi}}{dk} \right| > \eta_2$ where η_1 and η_2 are two positive scalars
236 defining the stopping criterion for the learning process, the learning criterion is not satisfied.
237 Therefore, the enrichment of the meta-model is done by computing $y_{new} = G(\mathbf{x}_{new})$ for a new
238 point \mathbf{x}_{new} . This \mathbf{x}_{new} is chosen close to the separator and far from other observations so that
239 its observation will bring valuable information. Therefore, it reads:

$$\mathbf{x}_{new} = \underset{\mathbf{x} \in U}{\operatorname{argmin}} \frac{s(\mathbf{x}) \max(d)}{d(\mathbf{x}) \max(s)} \quad (36)$$

240 where $s(\mathbf{x})$ is the distance between \mathbf{x} and the separator. $d(\mathbf{x})$ is the distance between \mathbf{x} and the
241 closest point in the DOE. Note that η_1 and η_2 are defined by the user. In this paper, we choose
242 $\eta_1 = 10^{-4}$ and $\eta_2 = 10^{-5}$, that is to say the same order of magnitude as in [12].

243 It results in the adaptive construction of the SVM classifier. Once the learning criterion is
244 reached, the Monte Carlo estimator is used to compute the probability of failure by classifying
245 the Monte Carlo population U . A second criterion on the COV of the probability is defined to
246 ensure the quality of the Monte Carlo estimator. If the criterion is not fulfilled, the Monte Carlo
247 population is enlarged to reduce the sampling error. Because of the poor convergence, a very
248 large Monte Carlo population can be required. In order to reduce the computational burden,
249 it is possible to use important sampling (used with Kriging in [32]), subset simulations [4] or
250 directional sampling [33, 34].

251 The resulting algorithm is called ASVM-MCS (Adaptive Support Vector Machine and Monte
252 Carlo Simulation).

253 4. Controlling the discretization error into SVM classifier meta-modeling

254 As explained in subsection 2.3, the use of finite element simulations introduces a discretization
255 error which can lead to a large error on the probability of failure. In this section, we propose
256 two strategies to benefit from the discretization error bounds on G_{ex} developed in subsection
257 2.3.2 to improve the estimation of the probability of failure *via* SVM classification. Two new
258 algorithms based on ASVM-MCS are detailed respectively in the next two subsections.

259 4.1. Computation of bounds on the probability of failure

260 In this subsection, we propose a methodology to compute error bounds on the probability
261 of failure using a unique mesh size. Indeed, we suggest to build two SVM classifiers. The first
262 classifier D^+ is built from the upper bounds observations $y = \operatorname{sign}(G^+(\mathbf{x}))$. Once this classifier
263 D^+ is built, it can be used to separate the Monte Carlo population into two sub-populations:

264 the guaranteed failure population $D_{pop,gf} = \{(\mathbf{x}_i)_{i=1..n_{MC}} | D^+(\mathbf{x}_i) = -1\}$ and its complementary.
 265 Indeed, evaluating $D^+(\mathbf{x}_i)$ for $i = 1..n_{MC}$ is cheap once the classifier D^+ is defined. The second
 266 classifier D^- is built from the lower bound observations $y = \text{sign}(G^-(\mathbf{x}))$. Once this classifier
 267 D^- is built, it can be used to separate the Monte Carlo population into two sub-populations:
 268 the guaranteed safe population $D_{pop,gs} = \{(\mathbf{x}_i)_{i=1..n_{MC}} | D^-(\mathbf{x}_i) = +1\}$ and its complementary.
 269 We propose to use these two classifiers to compute two probabilities $P^- = \frac{\text{card}(D_{pop,gf})}{n_{MC}}$ and
 270 $P^+ = 1 - \frac{\text{card}(D_{pop,gs})}{n_{MC}}$ as illustrated in Algorithm 1. The learning function ξ^+ is the ratio
 271 between the number of points of U located inside the margin of the classifier D^+ over the size of
 272 U . The learning function ξ^- is the ratio between the number of points of U located inside the
 273 margin of the classifier D^- over the size of U .

274 Assuming that the two classifiers are correct, which means they are well-trained and do not
 275 misclassify points in the Monte Carlo population, we have :

$$D_{pop,gf} \subset \{\mathbf{x} \in U | \text{sign}(G_{ex}(\mathbf{x})) = -1\} \subset (U - D_{pop,gs}) \quad (37)$$

276 Finally, by assuming that the Monte Carlo is large enough so that it does not introduce an
 277 approximation error, we can obtain bounds on the exact probability of failure:

$$P^- \leq P_{f,ex} \leq P^+ \quad (38)$$

278 The authors are aware that those bounds are not guaranteed because of the meta-modeling error
 279 (due to the construction of the SVM classifier) and of the approximation error (due to the finite
 280 size of the Monte Carlo population which is classified by the metamodel). However, those two
 281 errors are controlled : the learning criterion identifies new observations to update the meta-
 282 model and the coefficient of variation indicates if a larger Monte Carlo population is required.
 283 Directional sampling or important sampling may help to reduce the computational time if it is
 284 necessary. The bounds P^+ and P^- enables to know the influence of the discretization error on
 285 the probability of failure. Moreover, the two classifiers D^+ and D^- can be used to exhibit the
 286 population for which the mesh size is not fine enough to obtain a classification not polluted by
 287 the discretization error. Indeed, the set $D_{pop,uc} = \{\mathbf{x}_i \in U | D^-(\mathbf{x}_i) < 0 \text{ and } D^+(\mathbf{x}_i) > 0\}$, defined
 288 as the uncertain population, can be easily obtained from the two classifiers. If the bounding on
 289 the exact probability of failure is too large, the user can perform a classification only on $D_{pop,uc}$
 290 with a finer mesh.

291 4.2. Construction of multi-fidelity SVM-based meta-model

292 In this subsection, we propose a methodology to train a SVM classifier from observations
293 computed on different mesh sizes. Let us introduce a family of mesh sizes $(h_j)_j$ for $j = 0..M$
294 where h_0 corresponds to the coarsest mesh and h_M to the finest mesh. The key ingredient of
295 this method is the following : only observations G^m (defined in Equation (19)) for which the
296 sign is guaranteed by the bounds G^+ (defined in Equation (22)) and G^- (defined in Equation
297 (21)) are used to train the classifier. Note that using $G^m = G^m(\mathbf{x})$ is a better choice than using
298 the solution of the finite element problem $G_H(\mathbf{x})$ because $G_H(\mathbf{x})$ may not be in the interval
299 $[G^-; G^+]$. Therefore, the classifier is built only from observations that are not polluted by the
300 discretization error. If an observation at the point \mathbf{x} is such that $G^+G^- < 0$, then the simulation
301 is done at the same point but with a finer mesh as long as the state of the point (fail or safe)
302 is not guaranteed. The construction of the SVM classifier and the estimation of the probability
303 of failure is unchanged. As a consequence, the SVM classifier is built from computations done
304 on different mesh sizes. This is the reason why this approach builds a multi-fidelity SVM-based
305 meta-model. The algorithm of this method is given in algorithm 2. Note that a finest mesh is
306 defined so that the algorithm does not exhibit an infinite while loop. In that only case, it is
307 allowed to use the sign of G^m computed on the finest mesh to enrich the meta-model even if
308 $G^+G^- < 0$.

309 Note that during the execution of this algorithm, the upper and lower bounds are computed.
310 Therefore, it is possible, for the user, at the end of the algorithm, to build the SVM classifiers
311 from the collection of upper bounds and the collection of lower bounds, in a similar way as it is
312 done in the first subsection and therefore obtain bounds on the probability of failure.

313 5. Numerical experiments

314 In this section, we illustrate the two algorithms described in Section 4 on two numerical
315 examples. Both examples are two-dimensional mechanical problem with 2 random variables.
316 The first one is a square plate for which the exact solution is known. Therefore, for this problem,
317 the exact probability of failure is available. The second example is a cracked plate for which the
318 failure occurs when crack grows, according to Griffith criterion. For both problem the criterion
319 to stop the enrichment of the Monte Carlo population is $\eta_3 = 0.02$ and the criteria to stop the
320 learning of the SVM meta-model are $\eta_1 = 10^{-4}$ and $\eta_2 = 10^{-5}$. The initial size of the DOE is
321 set to $n_{DOE} = 12$. The computations were done on Dell mobile Precision 3530 CTO-Base with
322 Intel Core i7 8750H (6C, 2.2 4.1 GHz, 9 Mo, 45W).

323 5.1. Square plate

324 Let us consider a rectangular plate of dimension $3mm \times 3mm$ with the hypothesis of small
 325 perturbations and plane strain. All displacements are imposed to be null on the perimeter of the
 326 plate. The only external action is the force \underline{f} defined such that the exact solution is:

$$\underline{u}_{ex} = x(x-3)y(y-3) \left((y-3)^2 \underline{e}_1 + y \underline{e}_2 \right) \quad (39)$$

327 The solicitation S is the average of the component σ_{xy} of the stress on a square of size $0.5mm$
 328 located on the top left-hand corner of the structure, as illustrated in Figure 1 left.

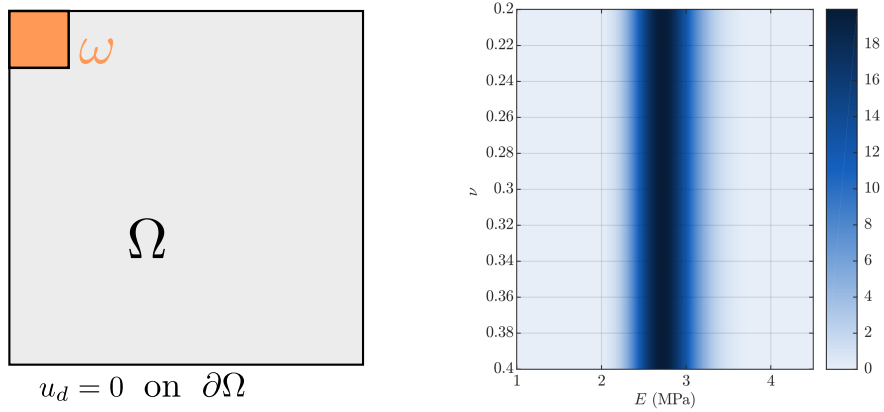


Figure 1: Left figure: square plate, Right figure: joint probability distribution of E and ν

329 The loading of the adjoint problem is therefore defined through the following linear form:

$$\tilde{L}(\underline{u}) = \frac{1}{mes(\omega)} \int_{\omega} (\underline{e}_1 \otimes \underline{e}_2) : \underline{\underline{\varepsilon}}(\underline{u}) d\omega \quad (40)$$

330 E is the Young modulus and follows a log-normal law of parameters $\mu = 1$ and $\sigma = 0.01$. ν
 331 is the Poisson ratio and follows a uniform law between $\nu_{min} = 0.2$ and $\nu_{max} = 0.4$. A map of
 332 density for the joint probability density function is plotted on Figure 1 right. Those two random
 333 variables are considered independent. The resistance R is deterministic and $R = 6.3MPa$. Since
 334 the exact displacement field is available, the exact probability of failure can be computed from
 335 (23): we obtain $P_{f,ex} = 1.1118 \cdot 10^{-4}$. Moreover, it is also possible to plot the exact limit
 336 state ($G_{ex} = 0$). Note that for this problem the exact limit state is linear. However, since the
 337 form of the limit state is usually unknown we used the kernel trick to build the SVM classifier
 338 which explains the form of the computed limit states. The finite element problem is solved with
 339 second-order elements composing the uniform mesh of size h .

340 *5.1.1. Strategy 1 : Computation of bounds thanks to 2 classifiers*

341 The strategy described in Algorithm 1 is applied for this example on a first mesh of size
342 $h_1 = 0.25\text{mm}$. In Table 1, we give the number of calls to the FE code as well as the bounds on
343 the probability of failure obtained for 5 different Monte Carlo populations. We observe that $P_{f,ex}$
344 lies between the computed bounds for all the Monte Carlo populations. For each simulation, the
345 two classifiers enable to exhibit the uncertain population. We also observe that the number of
346 calls is highly dependant on the Monte Carlo population. Then, a second mesh size $h_2 = 0.1\text{mm}$
347 is defined and the algorithm is run again to classify the uncertain population into certain safe
348 population and certain failure population. It results in additional calls to the finite element
349 solver but also in an improvement of the bounds on the exact probability of failure that can
350 be observed in Table 1. One can notice that for the first Monte Carlo population, the upper
351 $P_+ = 1.1115 \cdot 10^{-4}$ is smaller than $P_{f,ex} = 1.1118 \cdot 10^{-4}$. This is probably due to the finite size
352 of the Monte Carlo population as the requirement on the coefficient of variation is that it has
353 to be smaller than 0.02. In Figure 2, we plot for each mesh size, the two classifiers as well as
354 the certain failure domains and the certain safety domains for the first Monte Carlo population.
355 The exact limit state $G_{ex} = 0$ is also plotted in blue line on these plot. In addition, the points of
356 the Monte Carlo population that were selected for a finite element call to enrich the meta-model
357 are specified. We observe that the calls on the fine mesh h_2 are close to the exact limit state.
358 This is not surprising as the second mesh was used to classify the uncertain population. We also
359 notice in the right-hand side figure (zoom) that the exact limit state lies between the two limit
360 classifiers for both h_1 and h_2 .

361 *5.1.2. Strategy 2 : multi-fidelity classifier*

362 The strategy described in Algorithm 2 is applied on this example with a family of two meshes
363 of size $h_{max} = h_1$ and $h_{min} = h_2$. In Table 2, we give the number of calls to the FE code as
364 well as the probabilities of failure obtained for 5 different Monte Carlo populations. Note that
365 these populations are the same as the ones used to illustrate the first strategy. We observe that
366 the 5 probabilities of failure are very close to the exact probability of failure $P_{f,ex}$. In Figure
367 3, we plot the limit state as well as the failure and safety domains for the first Monte Carlo
368 population. The exact limit state $G_{ex} = 0$ is also plotted in blue line. In addition, the points of
369 the Monte Carlo population that were selected for a finite element call to enrich the meta-model
370 are specified. We observe that the calls on the fine mesh h_{min} are done close to the exact limit
371 state. The computational cost is focused on the frontier between safe and failure domains which

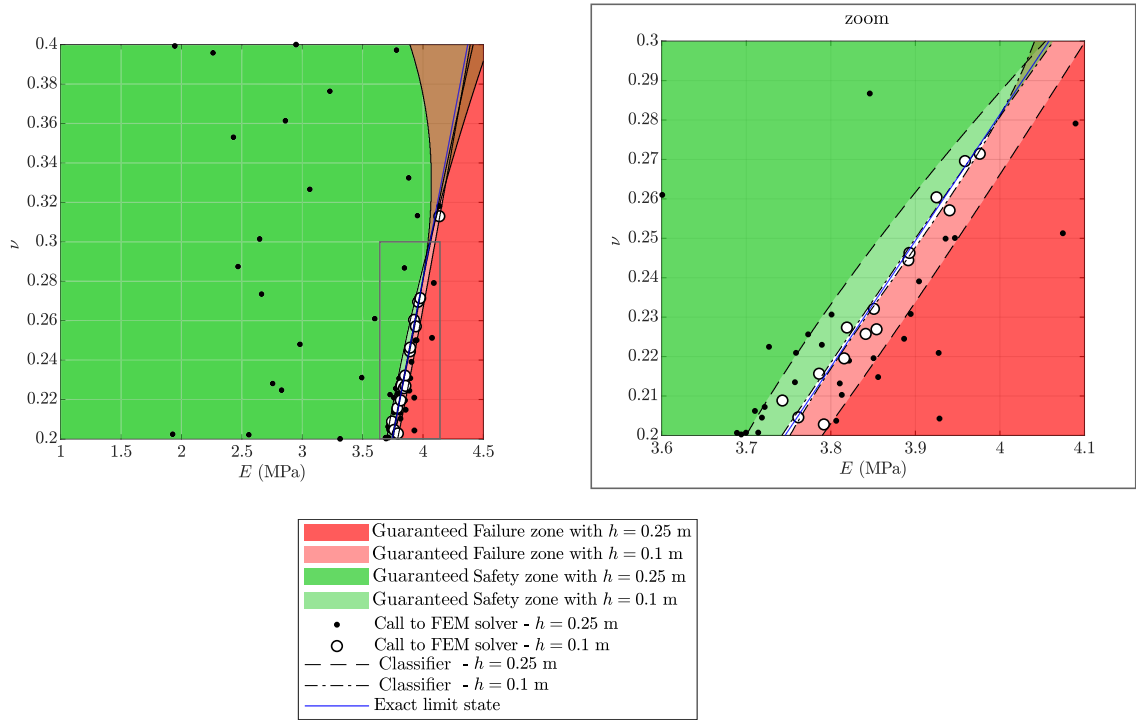


Figure 2: Left: Classifiers obtained for the strategy 1 on the square plate problem; Right: zoom

372 enables to obtain a good estimation of the probability of failure.

373 5.2. Cracked plate

374 Let us consider a cracked plate of length $L = 16mm$ and width $w = 7mm$. The length of
 375 the crack is denoted by a and is random. The material is linear elastic isotropic and the Poisson
 376 ratio is $\nu = 0.3$. The traction force \vec{F} of norm $F = 1MPa$ is applied on two sides of the plate
 377 with an angle θ , which is the second random variable of the problem. The solicitation S is the
 378 stress intensity factor on mode I, which we noted K_I , that can be computed thanks to integrals
 379 on a crown around the crack tip, see [35], as illustrated on Figure 4 left. The outer radius of the
 380 crown is 1.5mm and the inner radius is 1mm. The failure is defined from the Griffith criterion,
 381 therefore:

$$G = K_{lim} - K_I \quad (41)$$

382 where $K_{lim} = 22MPa\sqrt{mm}$ is the deterministic resistance. K_I is a linear quantity of interest
 383 with respect to the displacement. As a consequence, it is possible to define an adjoint problem,
 384 see [36], to obtain error bounds on G_{ex} , as explained in Section 2.3. Both random variables follow

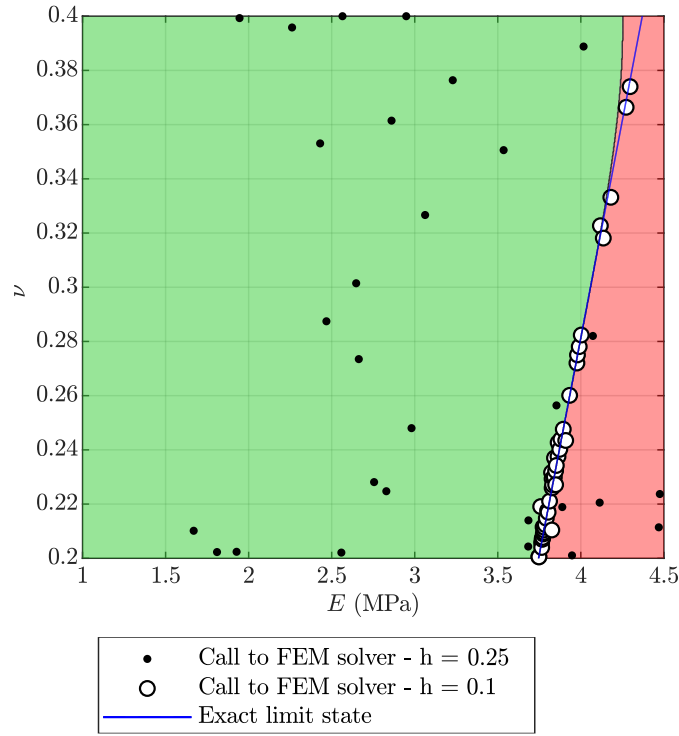


Figure 3: Limit state obtained for the strategy 2 on the square plate problem

385 the Beta distribution and are independent. The parameters given in Table 3 are the distribution
 386 of these two random variables which is illustrated in Figure 4 right.

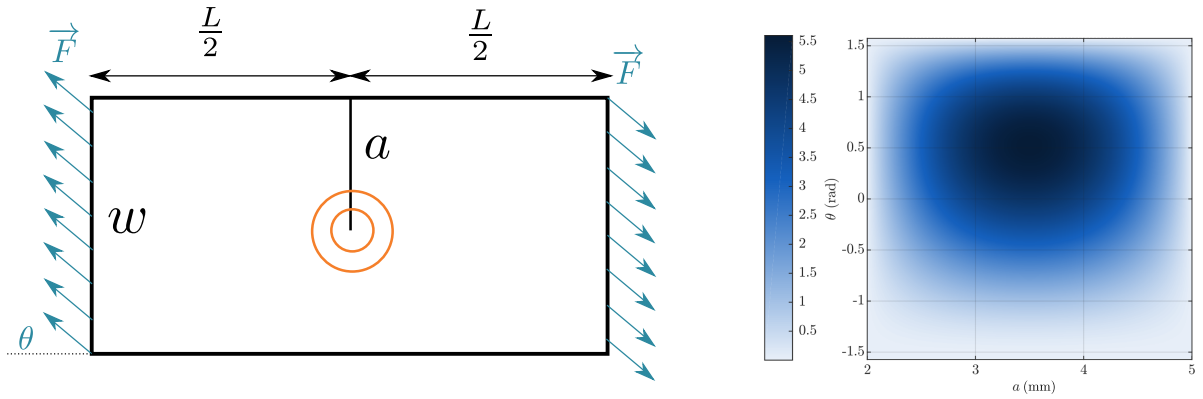


Figure 4: Left figure: cracked plate, Right figure: joint distribution distribution of a and θ .

387 For this problem, the exact value of the stress intensity factor for the first mode (crack
 388 opening) is not available for $\theta \neq 0$. A reference probability of failure P_{ref} is computed on a

389 overkill uniform mesh of size $h_{overkill} = 0.02mm$ with the reference method ASVMMCS [12]
 390 using the first Monte Carlo population. We obtained $P_{ref} = 5.9 \cdot 10^{-3}$. This simulation required
 391 69 calls to the finite element solver that last 16 184 seconds. The overkill classifier obtained
 392 during this process is plotted in blue on Figures 5 and 6.

393 5.2.1. Strategy 1

394 We applied the Algorithm 1 first with the mesh size $h_1 = 0.28mm$ and then with the mesh
 395 size $h_2 = 0.1mm$. In Table 4, we give the number of calls to the FE code as well as the bounds
 396 on the probability of failure obtained for 5 different Monte Carlo populations. The values of P^+
 397 and P^- are comparable for all the Monte Carlo populations. Similarly to the previous example,
 398 the number of calls highly depends on the Monte Carlo population. As for the first example, we
 399 observe that the bounds on the probability of failure are improved when the mesh size h_2 is used.
 400 In Table 5, we give the CPU time spent for the FE solving and the error estimation for the two
 401 mesh size h_1 and h_2 for 5 different Monte Carlo populations. We observe that the discretization
 402 error estimation is very expensive. However, this procedure enables to obtain bounds on the
 403 probability of failure, which is more informative than a unique value. In Figure 5, we give the
 404 classifiers as well as the guaranteed fail and safe domains for the two meshes for the first Monte
 405 Carlo population. The calls carried out to enrich the SVM-meta-models are also depicted in
 406 black or white, depending on the mesh size. We observe that the overkill classifier lies between
 407 the two classifiers in the uncertain white zone. Calls on the fine mesh are done close to the limit
 408 state. This figure clearly illustrates that applying again the Algorithm on a finer mesh enables
 409 to reduce the size of the uncertain zone.

410 5.2.2. Strategy 2

411 We applied the second algorithm on the cracked plate case for two mesh sizes $h_{max} = 0.28mm$
 412 and $h_{min} = 0.1mm$. In Figure 6, we represent the limit state obtained during the algorithm that
 413 separates the failure domain from the safety domain for the first Monte Carlo population. In this
 414 figure, the initial design of experiment and the learning points are presented. Points for which
 415 the coarse mesh was precise enough are in black. Points for which the sign of the performance
 416 function was not guaranteed and that required an additional call on the fine mesh are in white.
 417 We observe that far from the classifier, the coarse mesh is satisfying. However, as expected, when
 418 precision is required close to the limit state, calls on the fine mesh are done. We also observe that
 419 the multifidelity classifier (in black) is close to the overkill classifier (in blue) which illustrates
 420 the precision of the meta-model.

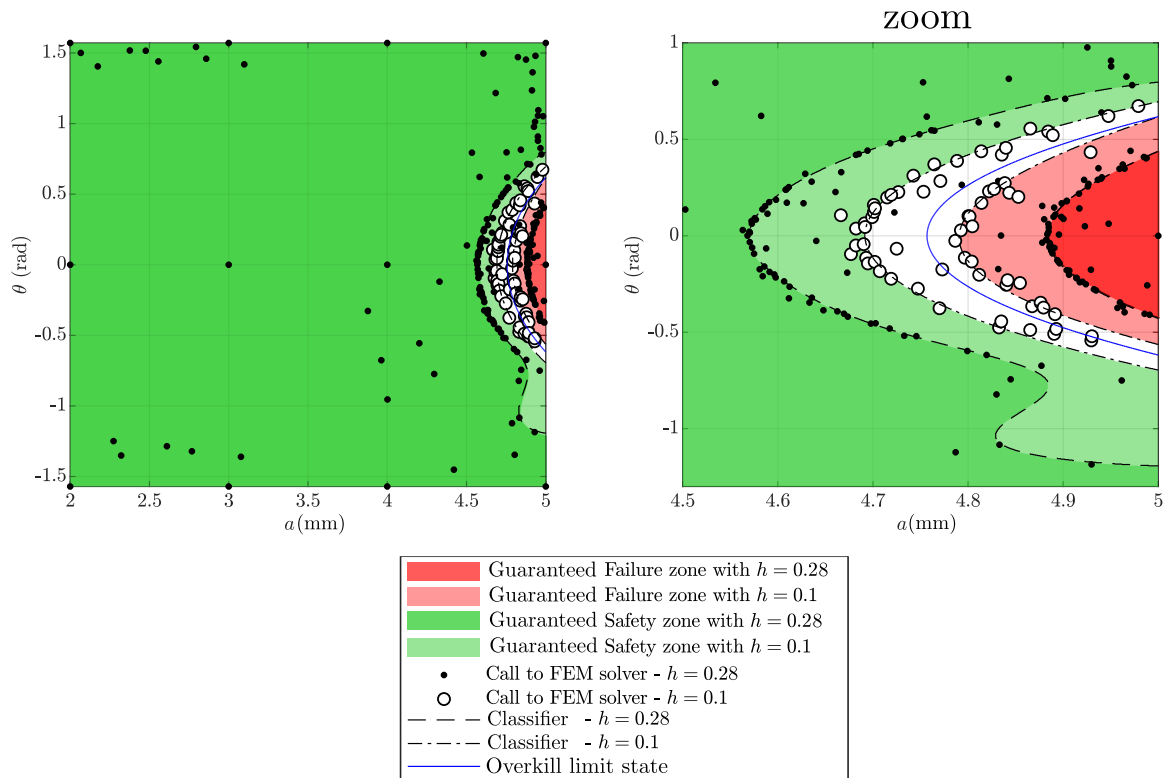


Figure 5: Limit states obtained for the strategy 1 on the cracked plate

421 In Table 6, we give the number of calls on each mesh, as well as the CPU time and the
422 probabilities of failure obtained for 5 Monte Carlo populations. We observe that for all the
423 Monte Carlo populations, the computed probability of failure has the same order of magnitude
424 as P_{ref} . The probabilities of failure are always larger than P_{ref} . This difference can be explained
425 by the fact that the computations were done on different meshes since $h_{min} \neq h_{overkill}$. Moreover,
426 for this second strategy, G^m is used whereas for the overkill monofidelity strategy, the FE output
427 G_H is used. From this table, we can observe that the computational cost associated to error
428 estimation is very large. This is due to the construction of the statically admissible stress field.
429 For the first Monte Carlo population, we observe that the CPU time for FE solving and error
430 estimation is equivalent to the CPU time for FE solving on the overkill mesh.

431 On this example, the second strategy requires less FE calls and therefore less CPU time than
432 the first strategy.

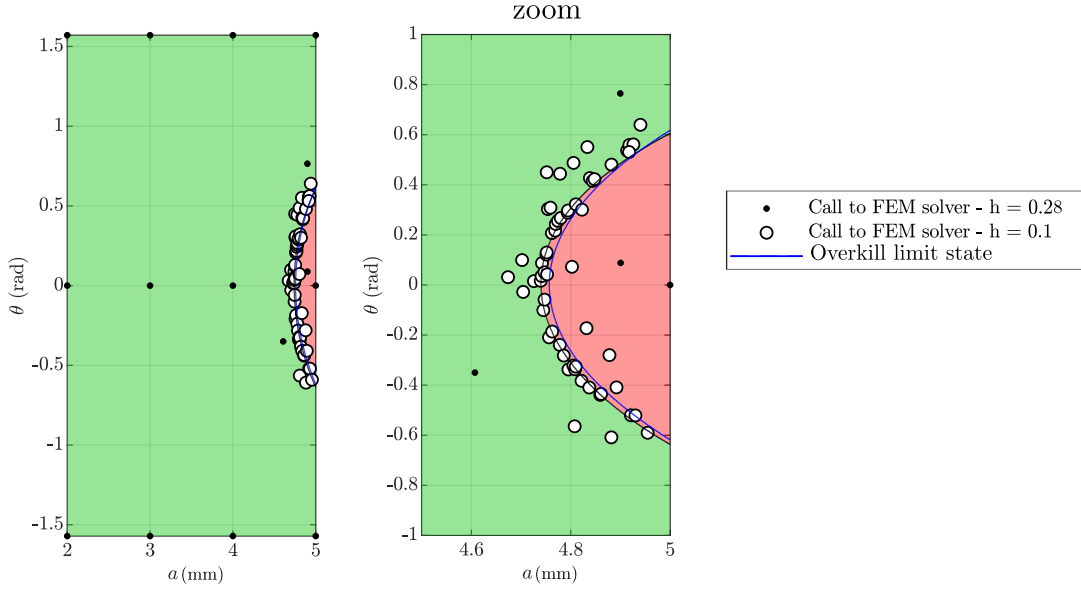


Figure 6: Left: limit state obtained for the strategy 2 on the cracked plate; Right: zoom

433 6. Conclusion

434 In this paper, we propose to use discretization error estimators to build an improved SVM-
 435 based meta-model for the estimation of the probability of failure for quasi-static mechanical
 436 problems. This first method considers a defined mesh size and enables to compute upper and
 437 lower bounds of the exact probability of failure. Two classifiers are built during the execution of
 438 the algorithm. It allows to identify the points in the Monte Carlo population for which the status
 439 (fail or safe) is polluted by the finite element discretization error. If this subpopulation is too
 440 large, that is to say if the bounds on the probability of failure are not precise enough, the user can
 441 run again this algorithm on this subpopulation with a finer mesh. The second method consists in
 442 using observations to construct the SVM classifier only if the discretization error bounds on the
 443 performance function enable to guarantee its sign. The SVM meta-model is therefore built from
 444 observations on different meshes. It results in a multi-fidelity meta-model. Those two strategies
 445 were illustrated on two mechanical examples and both strategies exhibit their ability to precisely
 446 estimate the probability of failure. Both strategies lead to an additional cost as they rely on
 447 discretization error estimations which can be expensive. However, the first strategy enables to
 448 obtain bounds on the probability of failure which is a more valuable information than a unique
 449 value. Moreover, the second strategy usually gives better results than monofidelity approach with

450 no error estimation as G^m is usually closer to G_{ex} than G_H . Finally, using purely monofidelity
451 meta-model would require to do a convergence study on the probability of failure with respect to
452 the mesh size. Therefore, the computational cost of our proposed methods should be compared
453 to the total cost of the whole convergence study. The two proposed approaches were illustrated
454 on problems with two random variables but the algorithms can be used straightforwardly for
455 problems with more random variables.

456 Further work will consist in exploiting the error map provided by the discretization error
457 estimators to build an optimal non uniform mesh. Another interesting topic would be to adapt
458 the stopping criterion of learning process and the stopping criterion on the size of the Monte
459 Carlo population to the discretization error. For example, in the first method, reaching a very
460 small COV for the probability of failure may not be relevant if the discretization error is too large.
461 This would require separating the different sources of error in the estimation of the probability
462 of failure.

463 7. Acknowledgment

464 This work was carried out within the project MUSCAS (MULTi-SCALE Stochastic computation
465 for MRE) granted by WEAMEC, West atlantic Marine Energy Community with the support of
466 Région Pays de la Loire and in partnership with Chantiers de l'Atlantique.

467 8. Bibliography

468 References

- 469 [1] P. Ciarlet, The Finite Element Method for Elliptic Problems, Studies in Mathematics and
470 its Applications, Elsevier Science, 1978.
471 URL <https://books.google.fr/books?id=TpHfoXnpKvAC>
- 472 [2] N. Metropolis, S. Ulam, The monte carlo method, Journal of the American statistical asso-
473 ciation 44 (247) (1949) 335–341.
- 474 [3] M. Lemaire, A. Chateauneuf, J.-C. Mitteau, Fiabilité des structures.
- 475 [4] S.-K. Au, J. Beck, Estimation of small failure probabilities in high dimensions by subset
476 simulation, Probabilistic engineering mechanics 16 (4) (2001) 263–277.

- 477 [5] M. Giles, Multilevel monte carlo path simulation, *Operations Research* 56 (3) (2008) 607–
478 617.
- 479 [6] M. Rashki, A. Ghavidel, H. Arab, S. Mousavi, Low-cost finite element method-based reli-
480 ability analysis using adjusted control variate technique, *Structural Safety* 75 (2018) 133–142.
- 481 [7] J. N. Fuhg, A. Fau, U. Nackenhorst, State-of-the-art and comparative review of adaptive
482 sampling methods for kriging, *Archives of Computational Methods in Engineering* 28 (4)
483 (2021) 2689–2747.
- 484 [8] F. Schoefs, Sensitivity approach for modelling the environmental loading of marine struc-
485 tures through a matrix response surface, *Reliability Engineering & System Safety* 93 (7)
486 (2008) 1004–1017.
- 487 [9] R. Teixeira, M. Nogal, A. O’Connor, Adaptive approaches in metamodel-based reliability
488 analysis: A review, *Structural Safety* 89 (2021) 102019.
- 489 [10] V. Vapnik, *The nature of statistical learning theory*, Springer science & business media,
490 2013.
- 491 [11] T. Most, An adaptive response surface approach for structural reliability analyses based on
492 support vector machines, in: *Proceedings of the eleventh international conference on civil,*
493 *structural and environmental engineering computing*, BHV Topping, 2007.
- 494 [12] Q. Pan, D. Dias, An efficient reliability method combining adaptive support vector machine
495 and monte carlo simulation, *Structural Safety* 67 (2017) 85–95.
- 496 [13] H. Song, K. Choi, I. Lee, L. Zhao, D. Lamb, Adaptive virtual support vector machine
497 for reliability analysis of high-dimensional problems, *Structural and Multidisciplinary Op-*
498 *timization* 47 (4) (2013) 479–491.
- 499 [14] J.-M. Bourinet, F. Deheeger, M. Lemaire, Assessing small failure probabilities by combined
500 subset simulation and support vector machines, *Structural Safety* 33 (6) (2011) 343–353.
- 501 [15] A. Basudhar, S. Missoum, An improved adaptive sampling scheme for the construction of
502 explicit boundaries, *Structural and Multidisciplinary Optimization* 42 (4) (2010) 517–529.
- 503 [16] I. Babuška, W. Rheinboldt, Error estimates for adaptive finite element computations, *SIAM*
504 *Journal on Numerical Analysis* 15 (4) (1978) 736–754.

- 505 [17] L. Mell, V. Rey, F. Schoefs, Multifidelity adaptive kriging metamodel based on discretiza-
506 tion error bounds, *International Journal for Numerical Methods in Engineering* 121 (20)
507 (2020) 4566–4583. arXiv:<https://onlinelibrary.wiley.com/doi/pdf/10.1002/nme.6451>,
508 doi:10.1002/nme.6451.
509 URL <https://onlinelibrary.wiley.com/doi/abs/10.1002/nme.6451>
- 510 [18] A. Ghavidel, M. Rashki, H. Arab, M. Moghaddam, Reliability mesh convergence analysis by
511 introducing expanded control variates, *Frontiers of Structural and Civil Engineering* 14 (4)
512 (2020) 1012–1023.
- 513 [19] K. Alvin, Method for treating discretization error in nondeterministic analysis, *AIAA journal*
514 38 (5) (2000) 910–916.
- 515 [20] L. Morse, Z. Khodaei, M. Aliabadi, A multi-fidelity boundary element method for struc-
516 tural reliability analysis with higher-order sensitivities, *Engineering Analysis with Boundary*
517 *Elements* 104 (2019) 183–196.
- 518 [21] J. Yi, F. Wu, Q. Zhou, Y. Cheng, H. Ling, J. Liu, An active-learning method based on multi-
519 fidelity kriging model for structural reliability analysis, *Structural and Multidisciplinary*
520 *Optimization* (2020) 1–23.
- 521 [22] L. Gallimard, Error bounds for the reliability index in finite element reliability analysis,
522 *International journal for numerical methods in engineering* 87 (8) (2011) 781–794.
- 523 [23] L. Mell, V. Rey, F. Schoefs, Two multifidelity kriging-based strategies to control discretiza-
524 tion error in reliability analysis exploiting a priori and a posteriori error estimators, *Com-*
525 *puters & Structures* (2022) 106897doi:<https://doi.org/10.1016/j.compstruc.2022.106897>.
- 526 [24] W. Prager, J. Synge, Approximations in elasticity based on the concept of function space,
527 *Quarterly of Applied Mathematics* 5 (3) (1947) 241–269.
- 528 [25] P. Ladeveze, D. Leguillon, Error estimate procedure in the finite element method and ap-
529 plications, *SIAM Journal on Numerical Analysis* 20 (3) (1983) 485–509.
- 530 [26] N. Parés, P. Díez, A. Huerta, Subdomain-based flux-free a posteriori error estimators, *Com-*
531 *puter Methods in Applied Mechanics and Engineering* 195 (4-6) (2006) 297–323.

- 532 [27] F. Pled, L. Chamoin, P. Ladevèze, On the techniques for constructing admissible stress
533 fields in model verification: Performances on engineering examples, *International Journal*
534 *for Numerical Methods in Engineering* 88 (5) (2011) 409–441.
- 535 [28] V. Rey, P. Gosselet, C. Rey, Study of the strong prolongation equation for the construction
536 of statically admissible stress fields: implementation and optimization, *Computer Methods*
537 *in Applied Mechanics and Engineering* 268 (2014) 82–104.
- 538 [29] P. Ladevèze, Strict upper error bounds on computed outputs of interest in computational
539 structural mechanics, *Computational Mechanics* 42 (2) (2008) 271–286.
- 540 [30] L. Dioşan, A. Rogozan, J.-P. Pecuchet, Improving classification performance of support
541 vector machine by genetically optimising kernel shape and hyper-parameters, *Applied In-*
542 *telligence* 36 (2) (2012) 280–294.
- 543 [31] S. Tong, D. Koller, Support vector machine active learning with applications to text classi-
544 fication, *Journal of machine learning research* 2 (Nov) (2001) 45–66.
- 545 [32] B. Echard, N. Gayton, M. Lemaire, N. Relun, A combined importance sampling and kriging
546 reliability method for small failure probabilities with time-demanding numerical models,
547 *Reliability Engineering & System Safety* 111 (2013) 232–240.
- 548 [33] P. Bjerager, Probability integration by directional simulation, *Journal of Engineering Me-*
549 *chanics* 114 (8) (1988) 1285–1302.
- 550 [34] J. Nie, B. Ellingwood, Directional methods for structural reliability analysis, *Structural*
551 *Safety* 22 (3) (2000) 233–249.
- 552 [35] M. Stern, E. B. Becker, R. S. Dunham, A contour integral computation of mixed-mode
553 stress intensity factors, *International Journal of Fracture* 12 (3) (1976) 359–368.
- 554 [36] L. Gallimard, J. Panetier, Error estimation of stress intensity factors for mixed-mode cracks,
555 *International Journal for Numerical Methods in Engineering* 68 (3) (2006) 299–316.

Algorithm 1:

```

Generate the the Monte Carlo population  $U$  of size  $n_{MC}$  ;
Generate the Design Of Experiment (DOE) of size  $n_{DOE}$ ;
for  $i = 1..n_{DOE}$  do
    Evaluate  $G^m(\mathbf{x}_i)$ ,  $G^+(\mathbf{x}_i)$  and  $G^-(\mathbf{x}_i)$ ;
    Complete the training samples with  $y_i = \text{sign}(G^+(\mathbf{x}_i))$  and  $z_i = \text{sign}(G^-(\mathbf{x}_i))$ ;
end
Train the SVM classifier  $D^+$  from the training sample  $(y_i)_{i=1..n_{DOE}}$ ;
Train the SVM classifier  $D^-$  from the training sample  $(z_i)_{i=1..n_{DOE}}$ ;
Classify the Monte Carlo population thanks to the SVM classifier  $D^+$  into two populations: guaranteed failure population
 $D_{pop,gf} = \{(\mathbf{x}_i)_{i=1..n_{MC}} | D^+(\mathbf{x}_i) = -1\}$  and its complementary;
Classify the Monte Carlo population thanks to the SVM classifier  $D^-$  into two populations: guaranteed safe population
 $D_{pop,gs} = \{(\mathbf{x}_i)_{i=1..n_{MC}} | D^-(\mathbf{x}_i) = +1\}$  and its complementary;
Estimate the probability  $P^- = \frac{\text{card}(D_{pop,gf})}{n_{MC}}$  ;
Compute the coefficient of variation  $\text{COV}^- = \sqrt{\frac{1-P^-}{P^- n_{MC}}}$  ;
Compute the learning functions  $\xi^-$  and  $\hat{\xi}^-$ ;
Estimate the probability  $P^+ = 1 - \frac{\text{card}(D_{pop,gs})}{n_{MC}}$  ;
Compute the coefficient of variation  $\text{COV}^+ = \sqrt{\frac{1-P^+}{P^+ n_{MC}}}$  ;
Compute the learning functions  $\xi^+$  and  $\hat{\xi}^+$ ;
Initialize enrichment iteration number  $k=0$ ;
while  $\xi^+ > \eta_1$  or  $\hat{\xi}^+ > \eta_1$  or  $\left| \frac{d\hat{\xi}^+}{dk} \right| > \eta_2$  or  $\xi^- > \eta_1$  or  $\hat{\xi}^- > \eta_1$  or  $\left| \frac{d\hat{\xi}^-}{dk} \right| > \eta_2$  or  $\text{COV}^+ > \eta_3$  or  $\text{COV}^- > \eta_3$  do
    if  $\xi^+ > \eta_1$  or  $\hat{\xi}^+ > \eta_1$  or  $\left| \frac{d\hat{\xi}^+}{dk} \right| > \eta_2$  or  $\xi^- > \eta_1$  or  $\hat{\xi}^- > \eta_1$  or  $\left| \frac{d\hat{\xi}^-}{dk} \right| > \eta_2$  then
        if  $\xi^+ > \eta_1$  or  $\hat{\xi}^+ > \eta_1$  or  $\left| \frac{d\hat{\xi}^+}{dk} \right| > \eta_2$  then
            Set  $k=k+1$  ;
            Select the optimal next sample point  $\mathbf{x}_{new}$  using equation (36) with classifier  $D^+$ ;
            Evaluate  $G^m(\mathbf{x}_{new})$ ,  $G^+(\mathbf{x}_{new})$  and  $G^-(\mathbf{x}_{new})$  at this point;
            Complete the training sample with  $y_{new} = \text{sign}(G^+(\mathbf{x}_{new}))$ ;
            Complete the training sample with  $z_{new} = \text{sign}(G^-(\mathbf{x}_{new}))$ ;
        end
        if  $\xi^- > \eta_1$  or  $\hat{\xi}^- > \eta_1$  or  $\left| \frac{d\hat{\xi}^-}{dk} \right| > \eta_2$  then
            Set  $k=k+1$  ;
            Select the optimal next sample point  $\mathbf{x}_{new}$  using equation (36) with classifier  $D^-$ ;
            Evaluate  $G^m(\mathbf{x}_{new})$ ,  $G^+(\mathbf{x}_{new})$  and  $G^-(\mathbf{x}_{new})$  at this point;
            Complete the training sample with  $y_{new} = \text{sign}(G^+(\mathbf{x}_{new}))$ ;
            Complete the training sample with  $z_{new} = \text{sign}(G^-(\mathbf{x}_{new}))$ ;
        end
        Train the SVM classifier  $D^+$  from the training sample  $(y_i)$ ;
        Train the SVM classifier  $D^-$  from the training sample  $(z_i)$ ;
        Classify the Monte Carlo population thanks to the SVM classifier  $D^+$  ;
        Classify the Monte Carlo population thanks to the SVM classifier  $D^-$  ;
        Estimate the probabilities  $P^-$  and  $P^+$  ;
        Compute the coefficients of variation  $\text{COV}^-$  and  $\text{COV}^+$ ;
        Compute the learning functions  $\xi^-$ ,  $\hat{\xi}^-$ ,  $\xi^+$  and  $\hat{\xi}^+$ ;
    end
    else
        Enlarge the Monte Carlo population;
        Classify the Monte Carlo population thanks to the SVM classifier  $D^+$  ;
        Classify the Monte Carlo population thanks to the SVM classifier  $D^-$  ;
        Estimate the probabilities  $P^-$  and  $P^+$  ;
        Compute the coefficients of variation  $\text{COV}^-$  and  $\text{COV}^+$ ;
        Compute the learning functions  $\xi^-$ ,  $\hat{\xi}^-$ ,  $\xi^+$  and  $\hat{\xi}^+$ ;
    end
end

```

Algorithm 2:

```

Define a list of mesh sizes  $(h_j)_{j=0..N}$ ;
Generate the the Monte Carlo population  $U$  of size  $n_{MC}$  ;
Generate the Design Of Experiment (DOE) of size  $n_{DOE}$ ;
Set  $j = 0$ ;
for  $i = 1..n_{DOE}$  do
    Evaluate  $G^m(\mathbf{x}_i)$ ,  $G^+(\mathbf{x}_i)$  and  $G^-(\mathbf{x}_i)$  with the mesh size  $h_j$ ;
    while  $G^+(\mathbf{x}_i)G^-(\mathbf{x}_i) < 0$  and  $j < M$  do
         $j = j + 1$ ;
        Evaluate  $G^m(\mathbf{x}_i)$ ,  $G^+(\mathbf{x}_i)$  and  $G^-(\mathbf{x}_i)$  with the mesh size  $h_j$ ;
    end
    Complete the training sample with  $y_i = \text{sign}(G^m(\mathbf{x}_i))$ ;
end
Set  $j = 0$ ;
Train the SVM classifier  $D$  from the training sample;
Classify the Monte Carlo population thanks to the SVM classifier  $D$  into two populations: failure population
 $D_{pop,f} = \{(\mathbf{x}_i)_{i=1..n_{MC}} | D(\mathbf{x}_i) = -1\}$  and safety population  $D_{pop,s} = \{(\mathbf{x}_i)_{i=1..n_{MC}} | D(\mathbf{x}_i) = +1\}$ ;
Estimate the probability  $P_f = \frac{\text{card}(D_{pop,f})}{n_{MC}}$  ;
Compute the coefficient of variation  $\text{COV} = \sqrt{\frac{1-P_f}{P_f n_{MC}}}$  ;
Compute the learning functions  $\xi$  and  $\hat{\xi}$ ;
while  $\xi > \eta_1$  or  $\hat{\xi} > \eta_1$  or  $\left|\frac{d\hat{\xi}}{dk}\right| > \eta_2$  or  $\text{COV} > \eta_3$  do
    if  $\xi > \eta_1$  or  $\hat{\xi} > \eta_1$  or  $\left|\frac{d\hat{\xi}}{dk}\right| > \eta_2$  then
        Select the optimal next sample point  $\mathbf{x}_{new}$  using equation (36);
        Evaluate  $G^m(\mathbf{x}_{new})$ ,  $G^+(\mathbf{x}_{new})$  and  $G^-(\mathbf{x}_{new})$  with the mesh size  $h_j$ ;
        while  $G^+(\mathbf{x}_{new})G^-(\mathbf{x}_{new}) < 0$  and  $j < M$  do
             $j = j + 1$ ;
            Evaluate  $G^m(\mathbf{x}_{new})$ ,  $G^+(\mathbf{x}_{new})$  and  $G^-(\mathbf{x}_{new})$  with the mesh size  $h_j$ ;
        end
        Complete the training sample with  $y_{new} = \text{sign}(G^m(\mathbf{x}_{new}))$ ;
    end
    Train the SVM from the training sample;
    Classify the Monte Carlo population thanks to the SVM classifier  $D$  into two populations: failure domain  $D_{pop,f}$  and
    safety domain  $D_{pop,s}$ ;
    Estimate the probability  $P_f = \frac{\text{card}(D_{pop,f})}{n_{MC}}$  ;
    Compute the coefficient of variation  $\text{COV} = \sqrt{\frac{1-P_f}{P_f n_{MC}}}$  ;
    Compute the learning functions  $\xi$  and  $\hat{\xi}$ ;
else
    Enlarge the Monte Carlo population;
    Classify the Monte Carlo population thanks to the SVM ;
    Estimate the probability  $P_f = \frac{\text{card}(D_{pop,f})}{n_{MC}}$  ;
    Compute the coefficient of variation  $\text{COV} = \sqrt{\frac{1-P_f}{P_f n_{MC}}}$  ;
    Compute the learning functions  $\xi$  and  $\hat{\xi}$ ;
end
end

```

$h_1 = 0.25\text{mm}$			$h_2 = 0.1\text{mm}$		
Nb calls	$P_- (\times 10^{-4})$	$P_+ (\times 10^{-4})$	Nb calls	$P_- (\times 10^{-4})$	$P_+ (\times 10^{-4})$
54	0.7031	1.6822	16	1.0602	1.1115
113	0.6771	1.7963	18	1.0659	1.1610
153	0.6756	1.7931	13	1.0524	1.12469
114	0.6620	1.8171	13	1.0812	1.1561
92	0.6925	1.8453	10	1.0162	1.3097

Table 1: Bounds of the probability of failure for the strategy 1 on the square plate problem

$P_f (\times 10^{-4})$	Nb calls h_{max}	Nb calls h_{min}
1.0979	74	45
1.1194	70	42
1.1109	75	41
1.0964	85	47
1.1121	74	44

Table 2: Estimated probabilities of failure with the strategy 2 on the square plate problem

random variable	distribution	lower bound	upper bound	first shape parameter	second shape parameter
a	Beta	2	5	2	2
θ	Beta	$-\frac{\pi}{2}$	$\frac{\pi}{2}$	3	2

Table 3: Distribution of the random variables a and θ

h_1			h_2		
Nb calls	$P_- (\times 10^{-3})$	$P_+ (\times 10^{-3})$	Nb calls	$P_- (\times 10^{-3})$	$P_+ (\times 10^{-3})$
194	0.938	23.00	65	4.022	10.36
171	0.923	22.44	52	4.274	10.45
110	0.944	22.68	52	3.868	10.46
164	0.961	22.53	61	4.043	10.42
156	0.919	22.32	69	3.932	11.56

Table 4: Bounds of the probability of failure for the strategy 1 on the cracked plate

h_1		h_2	
$t_{EF}(s)$	$t_{err}(s)$	$t_{EF}(s)$	$t_{err}(s)$
223	5293	202	11069
244	6499	188	11580
151	3938	206	13017
289	7553	277	19177
303	7878	519	27717

Table 5: Numerical cost for strategy 1 for 5 Monte Carlo populations

$P_f (\times 10^{-3})$	Nb calls $h_{max} = 0.28$	Nb calls $h_{min} = 0.2$	$t_{EF}(s)$	$t_{err}(s)$
6.765	73	58	326	16253
6.849	71	56	389	20447
7.062	66	53	326	15190
6.640	90	74	414	17129
6.995	93	73	362	14922

Table 6: Strategy 2 on the cracked plate for 5 Monte Carlo populations