Optimisation of interacting particle systems for rare event estimation

Jérôme Morio\textsuperscript{a,*,} Damien Jacquemart\textsuperscript{a,b}, Mathieu Balesdent\textsuperscript{a}, Julien Marzat\textsuperscript{a}

\textsuperscript{a} Onera - The French Aerospace Lab, BP 80100, 91123 Palaiseau Cedex, France
\textsuperscript{b} INRIA Rennes, ASPI Applications of Interacting Particle Systems to Statistics Campus de Beaulieu, 35042 Rennes, France

\textbf{A R T I C L E  I N F O}

Article history:
Received 29 October 2012
Received in revised form 19 March 2013
Accepted 25 March 2013
Available online 1 April 2013

Keywords:
Rare event
Probability
Interacting particle system
Optimisation
Hyperparameter
Surrogate model
Input–output model
Kriging

\textbf{A B S T R A C T}

The interacting particle system (IPS) is a recent probabilistic model proposed to estimate rare event probabilities for Markov chains. The principle of IPS is to apply alternatively selection and mutation stages to a set of initial particles in order to estimate probabilities or quantiles more accurately than with usual estimation techniques. The practical issue of IPS is the tuning of a parameter in the selection stage. Kriging-based optimisation strategy with a low simulation cost is thus proposed in order to minimise the probability estimate relative error. The efficiency of the proposed strategy is demonstrated on different test cases.

© 2013 Elsevier B.V. All rights reserved.

1. Introduction

Rare event estimation has become a large area of research in the reliability engineering and system safety domain. Importance sampling and importance splitting are the most well-known rare event simulation techniques for time-dependent complex systems. Importance sampling (Glynn and Iglehart, 1989; Juneja and Shahabuddin, 2001) consists in changing the initial probability distribution so that the rare event appears more frequently. The determination of an efficient auxiliary sampling density is far from obvious and often requires an optimisation stage and also some \textit{a priori} knowledge of the complex system. For a general Markov process with continuous state space, there does not seem to be a method that determines an efficient measure of importance sampling, because of the very broad variety of instances that can be involved. It is thus difficult to apply importance sampling in realistic cases for Markov processes with continuous state space.

The idea of importance splitting (also called subset simulation, subset sampling or sequential Monte Carlo) is to express the sought probability as a product of conditional probabilities that can be estimated within a reasonable computation time. These probability estimates are obtained by applying several stages of selection/mutation on an interacting particle population. The particles most likely to reach the rare event are duplicated (mutation stage) and the others are killed (selection stage). Particle filter algorithms also consider the same principles: the particles that suit best the observations are duplicated and the others are killed. This idea has been first proposed in a physical context in 1951 (Kahn and Harris, 1951), and numerous variants have been worked out since. Many particle models such as sampling branching models, excursion space exploration, branching particle systems, or related multi-level techniques have been proposed based on importance
sampling and splitting approaches. A detailed review of these algorithms is available in Del Moral and Lezaud (2006) and Del Moral (2004).

Interacting particle system (IPS) (Del Moral and Garnier, 2006) is considered here. It is a relatively new algorithm that improves the estimation of rare event probabilities based on evolutionary principles and importance sampling method. IPS is indeed equivalent to an importance splitting algorithm where the interacting particles have a resampling weight. The selection/mutation stages in IPS are nevertheless applied at different times of the Markov process evolution, unlike importance splitting where selection/mutation stages are carried out on the whole Markov trajectories. IPS can thus be interesting when compared to importance splitting if simulating the whole Markov trajectories is time-consuming. Moreover, an efficient IPS sampling weight can be determined through the definition of a selection function, whereas it is not the case for importance sampling. The theoretical efficiency of this approach has been demonstrated notably in Del Moral and Garnier (2006), Carmona et al. (2009) and Carmona and Crépey (2010) for the estimation of a rare event on a Markov chain at a finite time. The first application of interacting particle systems in the context of rare events arose in optical fibre communication (Garnier and Del Moral, 2006). Nevertheless, the practical applicability of IPS highly depends on the tuning of a hyperparameter which strongly influences the IPS convergence. Unfortunately, an initial guess of its value is not accessible for realistic applications such as in industrial cases.

The main existing tools for tuning hyperparameters are cross-validation and its variants (k-fold cross-validation, leave-one-out cross-validation, generalised cross-validation; see Golub et al., 1979). Cross-validation may be used to assess the performance level for a given value of the hyperparameter vector and then an optimisation procedure may be called upon to find the best tuning for these hyperparameters. In Kohavi and John (1995) and Hutter et al. (2007), such approaches have been exploited via a discretisation of the hyperparameter space. Bayesian networks have also been advocated in Pavon et al. (2008), by considering previous simulation runs as prior knowledge. Various other techniques have been employed for tuning purpose, such as Monte Carlo simulations (Gold and Sollich, 2003), neural networks (Korniienko et al., 2006) or evolutionary algorithms (Powell, 2002; Lin et al., 2008). In particular, the use of genetic algorithms should be mentioned with applications in control design (Varsek et al., 1993) or reliability assessment (Gen and Yun, 2006). Nevertheless, these approaches reveal to be computationally intensive, since they require a large exploration of the hyperparameter space. This may be prohibitive when the evaluation of the method performance is achieved via costly computer simulations, and even more when the number of those simulations is limited.

Kriging-based global optimisation (Santner et al., 2003) is employed here to provide optimal values of the hyperparameters at a limited computational cost. For example, these tools have been successfully applied to the optimal tuning of fault detection strategies in Marzat et al. (2010). The present paper reports the application of this strategy to the tuning of IPS algorithm, which does not seem to have been addressed in the literature so far. The principle of IPS is first presented and the impact of its tuning parameter on probability estimation is described on a simple case. Section 3 is devoted to the presentation of the automatic tuning methodology. Finally, the proposed optimisation strategy is applied in Section 4 on a realistic case of aircraft conflict probability estimation with very promising results.

2. Interacting particle system

2.1. General problem of rare event estimation

Rare event estimation can be characterised by a threshold exceedance of a real Markovian functional at deterministic time $n$. Consider a Markov chain $(X_n)_{n \in \mathbb{N}}$ with state space $E \subset \mathbb{R}^d$. It is assumed that the initial law $P(X_0 \in dx)$ and the transitions kernels $P(E_j \in dx', X_{j-1} = x)$ for $j = 1, \ldots, n$, have a density with respect to the Lebesgue measure, denoted by $\pi_0(x_0)$ and $\pi_j(x_j|x_{j-1})$. Hence, the density of the random trajectory $(X_0, \ldots, X_n)$ of the chain is given by

$$Q_n(x_0, x_1, \ldots, x_n) = \pi_0(x_0) \prod_{j=1}^n \pi_j(x_j|x_{j-1}). \quad (1)$$

The probability of interest is denoted by $P_n(A)$ and defined by

$$P_n(A) = P(V(X_n) \in A), \quad (2)$$

where $V$ is a function from $E$ to $\mathbb{R}$. Over the last few years, different algorithms (Bucklew, 2004; Cérou et al., 2006) have been proposed to estimate the probability $P_n(A)$ when this probability is rare ($< 10^{-4}$). Emphasis is put here on the IPS algorithm (Del Moral and Garnier, 2006; Carmona et al., 2009; Carmona and Crépey, 2010), which is one of the most efficient to solve this kind of problem. The following section reviews its principle.

2.2. Principle

For a discrete-time process, importance sampling algorithms (Bucklew, 2004) can help sampling rare events and improve the probability estimation in terms of variance. They consist in generating random weighted samples from an auxiliary distribution rather than the distributions of interest $\pi_j$. Nevertheless, in most cases, the studied system is sufficiently
complicated so that it is impossible for the user to determine properly a valuable auxiliary sampling distribution. Importance splitting (Cérou et al., 2006, 2005; Jasra et al., 2008; Carvalho and Lopes, 2007; Voutilainen and Kaipio, 2005) is another rare event estimation method. Its principle is to express the sought probability as a product of conditional probabilities that can be estimated within a reasonable computation time. However, it is not very adapted to the estimation of rare event in finite time. From past experience, IPS and splitting performances are generally similar.

The idea of IPS is to select and favour, at each iteration time \( j = 1, \ldots, n \), the Markov chains that are more likely to reach the rare set \( A \). The IPS algorithm consists in a set of \( N \) paths \((X^{(i)}_j)_{1 \leq j \leq n}, i = 1, 2, \ldots, N \). The initial generation is a set of \( N \) samples \( X^{(1)}_0, \ldots, X^{(N)}_0 \) independently generated from the initial distribution of the chain \( \pi_0 \). The trajectories are updated from \( j \) to \( j + 1 \) to advantage the ones that can potentially reach the rare event \( A \). The IPS algorithm is performed in two steps. First, the selection stage consists in choosing with replacement \( N \) paths according to an empirical weighted measure built with a function \( G_j \) that depends on \( X_{0j}, \ldots, X_{nj}^{(j)} \) where \( X_{0j}^{(i)} \) is the vector \((X^{(i)}_0, X^{(i)}_1, \ldots, X^{(i)}_j) \). The function \( G_j \) has to be positive and favour the paths that are more likely to reach the rare set. Secondly, the mutation stage consists in applying the Markov transition kernel \( \pi_{j+1} \) to the trajectory evolution. The complete algorithm is presented in the following:

(i) Set \( j = 0 \). Define the sample size \( N \) and the positive weight functions \( G_j \). Initialise weights to \( W^{(i)}_0 = \cdots = W^{(N)}_0 = 1 \).

(ii) Sample independently \( X^{(1)}_0, \ldots, X^{(N)}_0 \) from law \( \pi_0 \). A set of \( N \) particles \((X^{(i)}_0, W^{(i)}_0)\), \( i = 1, \ldots, N \), can be formed.

(iii) Estimate the normalising constant

\[
\eta_j = \frac{1}{N} \sum_{i=1}^{N} G_j(X^{(i)}_{0j}).
\]

(iv) Choose independently \( N \) paths according to the empirical distribution

\[
\mu_j(\mathrm{d}\tilde{X}, \mathrm{d}\tilde{W}) = \frac{1}{N \eta_j} \sum_{i=1}^{N} G_j(X^{(i)}_{0j}) \delta(x_i^{(i)} - W^{(i)}_0)(\mathrm{d}\tilde{X}, \mathrm{d}\tilde{W}).
\]

The selected particles are denoted by \((\tilde{X}^{(i)}_j, \tilde{W}^{(i)}_j)\) for \( i = 1, \ldots, N \).

(v) For \( i = 1, \ldots, N \), the particle \((\tilde{X}^{(i)}_j, \tilde{W}^{(i)}_j)\) is transformed into \((X^{(i)}_{j+1}, W^{(i)}_{j+1})\) in the following way:

\[
X^{(i)}_{j+1} \approx \frac{W^{(i)}_j}{\eta_j} \cdot G_j(\tilde{X}^{(i)}_{0j})^{-1}.
\]

(vi) If \( j < n \), then go back to stage (iii).

(vii) Estimate \( \hat{P}_\text{IPS}(A) \approx \hat{P}_\text{IPS} \) with

\[
\hat{P}_\text{IPS} = \frac{1}{N} \sum_{i=1}^{N} G_j(X^{(i)}_{0j}) \delta(x_i^{(i)} - W^{(i)}_0(1)).
\]

The IPS probability estimate is unbiased as demonstrated in Del Moral and Garnier (2006). The original paper Del Moral and Garnier (2006) also proposes to use one of the two following weighted functions \( G_j \), written here with respect to the present paper problem formulation:

\[
G^\beta_j(X_{0j}) = \exp(\beta V(X_j)) \quad \text{for some } \beta > 0, \tag{3}
\]

and

\[
G^\alpha_j(X_{0j}) = \exp(\alpha (V(X_j) - V(X_{j-1}))) \quad \text{for some } \alpha > 0. \tag{4}
\]

The function \( G^\beta_j \) often gives better results (Del Moral and Garnier, 2006) in practice and the function \( G^\beta_j \) will not be considered in the following. Note that using these functions does not require to store the whole trajectories \( X^{(i)}_{0j} \) in memory but only \( X^{(i)}_{0j} \) or \( X^{(i)}_{j-1} \) at each step.

IPS optimal tuning is nevertheless needed when one uses the functions \( G^\beta_j \) or \( G^\alpha_j \). In general, for two different probability estimations on the same Markov chain, the performance of IPS algorithm is different for the same values of \( \alpha \) (or \( \beta \)). So, although asymptotic variance expression can be computed (Del Moral and Garnier, 2006), optimal values of these parameters depend on the unknown probability to estimate.

2.3. Application and tuning problem

Let us consider the following toy case. The Markov chain considered is the Gaussian random walk \( X_{j+1} = X_j + W_{j+1} \) where the \( W_{j=1,\ldots,n} \) are i.i.d. (independent and identically distributed) Gaussian random variables with zero mean and variance 1.
and with $X_0 = 0$. One assumes in the following that $n = 16$ and one wants to estimate $P(X_n > 30)$. The IPS algorithm with weight function given in Eq. (4) is applied for different values of $\alpha$ and with $N = 2 \cdot 10^4$ particles. These estimations have been repeated 50 times to determine IPS estimate mean and relative error where the IPS relative error is defined by the ratio between IPS estimate standard deviation and IPS estimate mean. Fig. 1(a) shows the IPS probability estimate mean, and the relative error is given in Fig. 1(b). The theoretical probability is $3.18 \cdot 10^{-14}$. The IPS algorithm converges to the theoretical probability even if there is a high variability of the IPS results depending on the value of $\alpha$. Moreover, the choice of $\alpha$ depends on the sought probability. Indeed, the same experiments have been performed to estimate $P(X_n > 45)$. As shown in Fig. 2, the optimal parameter $\alpha$ is equal to 2.75, which means the tuning of $\alpha$ is only valuable for a given probability estimation. A wrong choice for this value could thus lead to erroneous results. In very simple cases, it is possible to determine theoretically the optimal value of $\alpha$ as shown in Del Moral and Garnier (2006) such as the Gaussian random walk. Nevertheless, in realistic situations, it is not possible to determine the optimal value $\alpha$ for the IPS algorithm and it is difficult to use IPS efficiently since it can lead to a strong misestimation. In industrial applications, it is not possible to sample as many Markov chains as in this simple case to tune the parameter $\alpha$. It is thus necessary to use an efficient optimisation algorithm with a low simulation cost. In the following, one proposes an optimisation methodology to determine valuable values of the IPS hyperparameter $\alpha$ at a relatively low computation cost.

3. Optimal determination of hyperparameters

The method proposed to determine an optimal IPS hyperparameter is based on design and analysis of computer experiments (DACE) methods. The objectives of DACE are as follows:

- to reduce the uncertainty in the experimental area, Gramacy and Lee (2009).
- to find extrema, Jones et al. (1998).
In this article, simulation is used to estimate the most efficient hyperparameter \( \alpha \) of the IPS system being simulated, that is, the hyperparameter \( \alpha \) that gives the lowest IPS relative error. A recent and efficient DACE method is considered for minimising the IPS relative error.

Consider the numerical simulation of a representative test case on which IPS to be tuned is applied. One can define a black-box function where the hyperparameter value \( \alpha \) is an input that should return as an output a scalar performance index \( s \). In the case of IPS, \( s \) is the IPS probability relative error. IPS probability relative error is defined by the ratio between IPS estimate standard deviation and IPS estimate mean. \( s \) is computed by repetitive estimations of the probability given by the IPS model. The hyperparameter \( \alpha \) is only assumed to belong to a known bounded set \( \mathbb{F} \). The tuning problem could then be formalised as the search for the optimal hyperparameter such that

\[
\tilde{\alpha} = \arg\min_{\alpha \in \mathbb{F}} s(\alpha).
\]

This is a difficult problem, since the only available information is the performance value at sampled locations of \( \alpha \). The proposed tuning methodology uses a Kriging model to approximate the unknown mapping from the hyperparameter space \( \mathbb{F} \) to the performance criterion \( s(\cdot) \). The so-called Efficient Global Optimization (EGO) algorithm (see Jones et al., 1998) is then employed to explore areas of the hyperparameter space that might lead to best performance and finally find an optimal tuning. The proposed strategy is now detailed.

Consider that a small number, \( m \), of possible \( \alpha \) tunings have already been experimented, resulting in the set \( A_m = \{\alpha_1, \ldots, \alpha_m\} \). The corresponding performance indices are gathered in the \( m \)-dimensional vector \( \mathbf{s}_m = [s(\alpha_1), \ldots, s(\alpha_m)]^T \).

Based on this initial knowledge, Kriging (Matheron, 1963) makes it possible to build a surrogate model \( \hat{S} \) of the black-box function between \( \alpha \) and \( s \), by modelling it as a Gaussian process \( S(\cdot) \) with mean function \( \mu(\cdot) \) and covariance function \( \text{cov}(\cdot, \cdot) \) (Lefebvre et al., 1996). One of the advantages of Kriging is the ability to estimate the variance \( \hat{\sigma}^2 \) of the prediction error, that is the confidence level on the surrogate model \( \hat{S} \). Kriging-based optimisation algorithms achieve a trade-off between local search (near the best known optimum) and global search (locations where the uncertainty on the surrogate is strong) (Jones, 2001). One of these strategies is the EGO procedure (Jones et al., 1998), which proceeds as follows:

(i) Sample \( m \) hyperparameter values \( \alpha \) to compute \( A_m \) and \( s_m \).
(ii) Fit a Kriging model on \( A_m \) and \( s_m \).
(iii) Find \( s_{\text{min}}^m = \min_{i=1,\ldots,m} \{s(\alpha_i)\} \).
(iv) Find \( \tilde{\alpha}_{m+1} = \arg\max_{\alpha \in \mathbb{F}} EI(\alpha, s_{\text{min}}^m, \hat{S}, \hat{\sigma}) \).
(v) Append \( \tilde{\alpha}_{m+1} \) to \( A_m \) and \( s(\tilde{\alpha}_{m+1}) \) to \( s_m \).
(vi) If \( m > m_{\text{max}} \), return \( s_{\text{min}}^m \) as the minimum relative error found.

Else, go to Step (ii) with \( m \leftarrow m + 1 \).

The working principle of this iterative algorithm is to replace the initial intractable optimisation problem (5) by the repeated optimisation of a much lighter function called the Expected Improvement (EI at Step (iv)), whose closed-form expression is defined by Schonlau (1997)

\[
\text{EI}(\alpha, s_{\text{min}}^m, \hat{S}, \hat{\sigma}) = \int_{-\infty}^{s_{\text{min}}^m} \left[ s_{\text{min}}^m - \hat{S} \right] h[\hat{S}] d\hat{S},
\]

where \( h[\hat{S}] \) is the distribution of \( \hat{S} \). EI assumes that this distribution is a Gaussian distribution with the estimated mean \( \hat{S} \) and a standard deviation equal to the estimated predictor standard deviation \( \hat{\sigma} \). The previous equation then becomes

\[
\text{EI}(\alpha, s_{\text{min}}^m, \hat{S}, \hat{\sigma}) = (s_{\text{min}}^m - \hat{S}(\alpha)) \psi \left( \frac{s_{\text{min}}^m - \hat{S}(\alpha)}{\hat{\sigma}(\alpha)} \right) + \hat{\sigma}(\alpha) \psi \left( \frac{s_{\text{min}}^m - \hat{S}(\alpha)}{\hat{\sigma}(\alpha)} \right),
\]

in which \( \psi \) and \( \Psi \) are respectively the probability density and cumulative distribution functions of the normal distribution. This function is computationally light, since it only involves the Kriging linear prediction and standard deviation. It could thus be optimised at each step via an auxiliary algorithm to be chosen (in our experiments, DIRECT Jones et al., 1993 was used). Thanks to the guidance of this sampling criterion, several optimal candidate points are explored iteratively, which provides in the end a set of appropriate hyperparameters \( \alpha \) for the application considered. The stopping condition of EGO is \( m_{\text{max}} \), which is the budget allotted for the black-box evaluations (it could also be expressed in terms of computation time). Note that other stopping criteria could also be considered, such as thresholds on the successive maximum values of EI obtained at Step (iv) (Schonlau, 1997).

Computation cost in simulation time or in IPS generated trajectories are highly correlated since the Kriging optimisation computation time can be neglected when compared to IPS simulation one. Considering time or IPS generated trajectories for computation cost is equivalent. The computational cost C in IPS generated trajectories of the proposed Kriging-based IPS...
can be derived in the following way. Indeed, there is \( m_{\text{max}} \) Kriging evaluations of \( s \) with the proposed strategy. Moreover, each estimation of the relative error \( s \) requires \( n_{\text{rep}} \) repetitive IPS estimations with \( N \) particles. The computational cost in IPS generated trajectories is then simply given by the following equation:

\[
C = m_{\text{max}} \times n_{\text{rep}} \times N.
\] (8)

In the following, two IPS runs have thus the same computation cost if the same number of IPS particle trajectories are generated. The parameter \( n_{\text{rep}} \) has been set to 50 for the different simulation cases but it can be decreased for computational reasons in realistic complex situations. Moreover, the proposed set \( \mathcal{F} \) is defined as \([0, 10]\) in the simulation cases. This interval is well-adapted to general situations, although low probabilities \((P_{\text{min}}(A) < 10^{-20})\) or small Markov chain length \((n < 5)\) may require an expansion of the set \( \mathcal{F} \).

### 4. Application of the proposed strategy

In this section, we apply the methodology proposed in the previous section to a toy case and a realistic situation of aircraft conflict.

#### 4.1. Toy case

The toy case described in Section 2.3 is considered with the optimisation strategy described in Section 3. The number \( m \) of initial values of \( \alpha \) where the IPS relative error is estimated is set to 5 in order to decrease the computation cost. These values are determined using random Latin Hypercube Sampling (LHS) (McKay et al., 1979). The number \( m_{\text{max}} \) of authorised \( \alpha \) samples is 15. The relative error \( s \) and the probability mean estimate are estimated with \( n_{\text{rep}} = 50 \) repetitive estimations of the probability of interest. An example of IPS relative error optimisation is given in Fig. 3. In this case, the algorithm finds an IPS minimum relative error equal to 22\% obtained for \( \alpha_{\text{min}} = 1.96 \) where \( s(\alpha_{\text{min}}) = s_{\text{min}}^{\alpha} \). This tuning value is very efficient and a low computation effort has been required to determine this approximated optimum. It is thus an improvement for the users of IPS algorithm since an initial guess of \( \alpha \) is not obvious. We repeated this optimisation 20 times with different initial LHS and estimated the average, worst and best performances of the optimisation strategy in Table 1. In the worst case of these 20 repetitions, the IPS relative error obtained with the proposed tuning is lower than 25\%.

It is also interesting to compare the results given by single runs of IPS without using the proposed Kriging based approach but with a large number of particles to keep the computational cost constant. In these IPS single run simulations, the number of particles is set to \( m_{\text{max}} \times N = 3 \cdot 10^5 \). The number of repeated simulations to estimate the probability and its relative error is kept to \( n_{\text{rep}} = 50 \). The corresponding probability and relative error estimate are given in Fig. 4. If the initial choice of \( \alpha \) is included in the interval \([1.15; 2.45]\), that is, if the chosen value of \( \alpha \) is not too far from its optimum value, then it is better in terms of relative error to consider IPS simulations with a high number of particles than the proposed Kriging-based IPS optimisation. For other values of \( \alpha \), the algorithm proposed in this article is more efficient.

#### 4.2. Aircraft trajectory model

A model for computing conflict probability estimation between two aircraft is presented in this section. It is based on realistic measurements (Paielli and Field, 1998) of 9500 aircraft flying at 29 000 ft over a period of 20 min following a straight line, with a constant nominal speed of 10 nmi/min (nmi = nautical miles) (Jacquemart and Morio, 2013).

An aircraft trajectory is modelled by a 2-dimensional stochastic process in discrete time defined by

\[
X_{j+1} = X_j + v h + \sqrt{h} \sigma h W_n,
\]

where \( X_j \) is the position of the aircraft at time \( j \), \( v \) is a two-dimensional speed vector, \( h \) is the discretisation time step, \( \sigma h \) is a squared correlation matrix and \( W_j \) is a standard two-dimensional Gaussian random variable. The initial position error is given by \( X_0 \sim N(0, A_0) \), for a given covariance matrix \( A_0 \).

The cross-correlation between the along-track and cross-track errors is small enough to be considered as null. The vertical position error can be negligible (Paielli and Erzberger, 1997) when compared to along-track and cross-track position errors, which is why the trajectory model considered is two-dimensional. Let us define \( X_j = (X_j^a, X_j^c) \) the aircraft position at time \( j \) with \( X_j^a \) the along-track aircraft position and \( X_j^c \) the cross-track aircraft position. If it is assumed that the aircraft has a

<table>
<thead>
<tr>
<th>Table 1</th>
<th>General results of hyperparameter ( \alpha ) optimisation on a toy case.</th>
</tr>
</thead>
<tbody>
<tr>
<td>( \alpha_{\text{min}} )</td>
<td>( s_{\text{min}}^{\alpha} ) (%)</td>
</tr>
<tr>
<td>Average result</td>
<td>1.93</td>
</tr>
<tr>
<td>Worst result</td>
<td>2.01</td>
</tr>
<tr>
<td>Best result</td>
<td>1.92</td>
</tr>
</tbody>
</table>
constant speed \( v \) in a coordinate system where the abscissa axis is parallel to aircraft trajectory, and taking into account the independence between the along-track and cross-track errors, one has

\[
\begin{align*}
X_{j+1}^a &= X_j^a + v h + \sqrt{h g^a(jh)} W_j^a \\
X_{j+1}^c &= X_j^c + \sqrt{h g^c(jh)} W_j^c,
\end{align*}
\]  

(9)
Fig. 4. IPS probability estimate mean and relative error for different values of $\alpha$, $P[X_n > 30]$ and 300 000 IPS particles.

where $h$ is the discretisation step and $W^a_j$ et $W^c_j$ are two independent standard one-dimensional Gaussian random variables, with initial conditions

$$\begin{align*}
X^a_0 &\sim \mathcal{N}(0, \nu^2_{0,a}), \\
X^c_0 &\sim \mathcal{N}(0, \nu^2_{0,c}).
\end{align*}$$

(10)

The initial conditions are experimentally set to Jacquemart and Morio (2013)

$$\nu_{0,a} = \nu_{0,c} = \nu_0 = 0.333.$$  

(11)

The functions $g^a$ and $g^c$ are defined as

$$g^a(x) = \sqrt{2(r^2_a x + r_a \nu_0)},$$  

(12)

and

$$g^c(x) = \sqrt{2(2p^3 x^3 + 3pqx^2 + (2p\nu_0 + q^2)x + 2q\nu_0 x)}$$

(13)

with $x \in \mathbb{R}^+$ and $r_a = 0.233$, $p = 0.0068$ and $q = 0.328$ (Jacquemart and Morio, 2013).

Now, if the speed vector has an angle $\theta$ with an horizontal axis, the aircraft position is obtained by applying a rotation matrix of angle $\theta$. One has then

$$X_{j+1} = X_j + R_\theta \begin{pmatrix} v h \\ 0 \end{pmatrix} + \sqrt{h}R_\theta \begin{pmatrix} g^a(jh) & 0 \\ 0 & g^c(jh) \end{pmatrix} W_j,$$

(14)

with $R_\theta = \begin{pmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{pmatrix}$. The term $X_0$ is defined by

$$X_0 = R_0 \begin{pmatrix} \nu_{0,a} & 0 \\ 0 & \nu_{0,c} \end{pmatrix} W_0,$$

(15)

where $W_0$ is the 2-dimensional standard Gaussian random variable.

4.2.1. Conflict probability between aircraft

If one considers two aircraft indexed by 1, 2, the two following stochastic processes define the aircraft trajectories:

$$\begin{align*}
X^1_{j+1} &= X^1_j + R_{\theta_1} \begin{pmatrix} v_1 h \\ 0 \end{pmatrix} + \sqrt{h}R_{\theta_1} \begin{pmatrix} g^a(jh) & 0 \\ 0 & g^c(jh) \end{pmatrix} W^1_j, \\
X^2_{j+1} &= X^2_j + R_{\theta_2} \begin{pmatrix} v_2 h \\ 0 \end{pmatrix} + \sqrt{h}R_{\theta_2} \begin{pmatrix} g^a(jh) & 0 \\ 0 & g^c(jh) \end{pmatrix} W^2_j.
\end{align*}$$

(16)

The process to be studied is the distance between the two aircraft defined by

$$D_j = \|X^1_j - X^2_j\|, \quad j \in \{0, \ldots, n\}.$$  

(17)
where $n$ is set in the proposed case to $n = 200$ which is equivalent to a period of 20 min. The probability of interest in this application is

$$
P(\exists j \in \{0, \ldots, n\}, D_j < \delta),$$

where $\delta$ is a specified distance threshold. For instance, setting $\delta$ as the sum of the two semi-wingspans of the aircraft corresponds to the probability of collision. When one talks about conflict probability, $\delta$ is set to 8 nmi in this article. Fig. 5 shows a sample of a couple of trajectories with a selected flight plan. The problem defined in Eq. (18) is not exactly the same as the one described in Section 2.1 but can be modified to fit in this formalism. The idea is to introduce the minimum process $M_j$ of the Markov chain $D_j$ before time $j$,

$$M_j = \min_{l=0, \ldots, j} D_l.$$  

The chain $M_j$ is not Markovian. It is thus necessary to apply IPS algorithm on the process $S_j$

$$S_j = (X_1^j, X_2^j, M_j),$$

which is Markovian and then to estimate the probability $P(M_n < \delta)$ since

$$P(M_n < \delta) = P(\exists j \in \{0, \ldots, n\}, D_j < \delta).$$

The IPS algorithm with weight function given in Eq. (4) is applied for different values of $\alpha$ and with $N = 2 \cdot 10^4$ particles to estimate the probability $P(M_n < \delta)$. These estimations have been repeated 50 times to determine IPS estimate mean and relative error $s$. Fig. 6(a) shows the IPS probability estimate mean and IPS relative error in % The probability estimated with a $5 \cdot 10^9$ Monte Carlo simulations is equal to $0.6 \cdot 10^{-8}$, to be used as a reference value. The IPS algorithm converges to this probability even if there is a high variability of the IPS results depending on the value of $\alpha$. A wrong choice
of this value could lead to erroneous results and moreover only a very small interval of $\alpha$ values gives valuable results. In this realistic case of aircraft conflict, it is impossible to determine theoretically an optimal value of $\alpha$ for the IPS algorithm which may reduce its applicability. If one uses the $\alpha$ optimisation strategy described in Section 3, one can obtain the results given in Fig. 7. The number $m$ of initial values of $\alpha$, where the IPS relative error is estimated, is 5 and they are determined using random LHS. The number $m_{\text{max}}$ of authorised $\alpha$ samples is still equal to 15. In this case, the algorithm finds an IPS minimum relative error equal to 51% obtained for $\alpha_{\text{min}} = 0.91$. This tuning value is thus very efficient. We repeated this optimisation 20 times with different initial LHS and estimated the average, worst and best performance of the optimisation strategy in
Table 2. General results of hyperparameter $\alpha$ optimisation on a realistic aircraft conflict.

<table>
<thead>
<tr>
<th></th>
<th>$\alpha_{\text{min}}$</th>
<th>$s_{\text{max}}$ (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Average result</td>
<td>0.89</td>
<td>59</td>
</tr>
<tr>
<td>Worst result</td>
<td>1.02</td>
<td>70</td>
</tr>
<tr>
<td>Best result</td>
<td>0.93</td>
<td>47</td>
</tr>
</tbody>
</table>

Fig. 8. IPS probability estimate mean and relative error for different values of $\alpha$ and 300 000 IPS particles.

Table 2. In the worst case of these 20 repetitions, the IPS relative error obtained with the proposed tuning is lower than 70%. It is the best result one can obtain with the IPS method for such a low probability estimate and this limited number $N$ of trajectory simulations. The only way to reduce the estimate relative error is no more the improvement of the $\alpha$ tuning but the increase of $N$.

The results given by single runs of IPS without using the proposed Kriging-based approach but with a large number of particles can be compared. The computational cost is also kept constant. In these IPS single run simulations, the number of particles is set to $m_{\text{max}} \times N = 3 \cdot 10^5$. The number of repeated simulations to estimate the probability and its relative error is kept equal to $n_{\text{rep}} = 50$. The corresponding probability and relative error estimate are given in Fig. 8. If the initial choice of $\alpha$ is included in the interval $[0.5 ; 1.3]$, then it is better in terms of relative error to consider IPS simulations with a high number of particles than the proposed Kriging-based IPS optimisation. For other values of $\alpha$, the algorithm proposed in this article is more efficient.

5. Conclusion

In this article, a Kriging-based strategy for estimating the optimal IPS hyperparameter has been proposed. This tuning is an issue since it is very complicated to determine a priori efficient values for $\alpha$. This is no more the case with the proposed method that enables a practical use of IPS since it does not require a high simulation cost. This algorithm has been applied successfully to the Gaussian random walk and to a realistic case of aircraft conflict estimation. A possible perspective to this work is the ability to modify the value of $\alpha$ during the IPS process in order to improve the estimation accuracy. For that purpose, there exist some alternative adaptive IPS models based on adaptive resampling procedures (Del Moral et al., 2012; Wolters, 2012). The idea is to use a sequence of weighted approximations up to the first time the product of sampling weights induce some degenerate estimate with respect to some criteria such as weight variance or entropy. It would be thus very interesting to connect the Kriging-IPS algorithm with these more classical resampling procedures.

Another perspective of this approach is to take into account the simulation error. The number of repetitions to estimate $s$ has been set to 50 in this article. Thus, the estimation of relative error is relatively accurate and taking simulation error into account will not change the results of this article. Nevertheless, if one wants to reduce the number of repetitions to estimate $s$ for computation time reason, it becomes necessary to consider the simulation error in the Kriging model. Mean and standard deviation of the relative error are still available since the relative error is estimated with Monte-Carlo estimations. The use of the Kriging model with noisy simulations has been reviewed in Picheny et al. (in press).

Acknowledgements

The work of Damien Jacquemart is financially supported by DGA (Direction Générale de l’Armement) and Onera, The French Aerospace Lab. The authors thank the two anonymous reviewers for their valuable remarks.
References


