Improving the Convergence of Iterative Importance Sampling for Computing Upper and Lower Expectations

Thomas Fetz THOMAS.FETZ@UIBK.AC.AT
Unit for Engineering Mathematics, University of Innsbruck, Austria

Abstract

The aim of this paper is to present methods for improving the convergence of an iterative importance sampling algorithm for calculating lower and upper expectations with respect to sets of probability distributions. Our focus here is on the reuse and the combination of results obtained in previous iteration steps of the algorithm.

Keywords: Monte Carlo simulation, importance sampling, reweighting, imprecise probability, lower/upper expectations, lower/upper probabilities.

1. Introduction

Our aim is to estimate the lower expectation

$$\theta_* = \min_{t\in\mathscr{T}} \int h(x) f_t(x)\,\mathrm{d}x \tag{1}$$

of a function h with respect to a parametrized family of probability density functions f_t, over all $t \in \mathscr{T}$. The case of upper expectations θ^* can be treated in similar way.

In joint work with M. C. M. Troffaes [9] we studied the convergence of an iterative importance sampling estimator developed in [2, 7] and based on earlier work [4, 3, 8] by formulating it as a fixed point of an operator. Here, we present methods to improve the convergence of the iterative importance sampling. These methods make use of results obtained in previous iteration steps of the algorithm by combining them. We think that these methods are complementary to the method of increasing the sample coverage addressed in [9] which also improves convergence of the fixed point iteration.

The plan of the paper is as follows: In Section 2 we introduce importance sampling and in Section 3 the iterative version following [9]. In Section 4 we show how to combine results of previous iteration steps to improve convergence and present an illustrative example.

2. Importance Sampling

Let f_t be a density parametrized by $t = (t_1,\dots,t_m) \in \mathscr{T}$. We are interested in estimating the lower expectation in (1) using Monte Carlo simulation and assume that samples from f_t can be generated as follows:

We start from a random variable V (e.g. uniform in $[0,1]^k$), and a function x_t of V, such that $x_t(V) \sim f_t$. For example, we have given the following: $t = (\mu,\sigma)$, $f_t \sim N(\mu,\sigma^2)$ and $V = (U_1,U_2)$ which is uniformly distributed on $[0,1]^2$. Then

$$x_t(V) = x_{(\mu,\sigma)}(V) = \mu + \sigma\sqrt{-2\ln U_1}\cos(2\pi U_2) \tag{2}$$

will have the desired Gaussian distribution [1]. The reason for making the function x_t explicit is that we need to control the randomness throughout the iterative importance sampling algorithm. We need to describe the sample itself as a deterministic function of the parameter t because otherwise the iteration would not work. That means that the sample points are transformed by parameter t and not newly generated when changing t. This can also be achieved in a very simple way: we only have to restart the random number generator with always the same seed when generating a new sample for a different parameter value t.

We start from an i.i.d. basic sample $\Omega = (V_1,V_2,\dots,V_n)$, to obtain the desired i.i.d. sample

$$x_s(V_1),\dots,x_s(V_n) \tag{3}$$

from f_s, for some fixed parameter $s \in \mathscr{T}$. Now, because

$$\int h(x) f_t(x)\,dx = \int \frac{f_t(x)}{f_s(x)} h(x) f_s(x)\,\mathrm{d}x, \tag{4}$$

we can use this sample from f_s to estimate the expectation of h with respect to f_t, for any $t \in \mathscr{T}$:

$$\hat{\theta}_{\Omega,s}(t) = \frac{1}{n}\sum_{i=1}^{n} w_{st}(x_s(V_i)) h(x_s(V_i)) \tag{5}$$

with weights

$$w_{st}(x) = f_t(x)/f_s(x). \tag{6}$$

We mention that this approach of using importance sampling or reweighting techniques can also be found in [5, 10, 11] and in [6] where it is called "what-if sampling".

Our estimator $\hat{\theta}_{\Omega,s}$ is based on a fixed set Ω of basic samples and on a fixed parameter s for which the sample points $x_s(V_i)$ are generated. Alternatively, the self-normalised importance sampling version can be used:

$$\hat{\theta}_{\Omega,s}(t) = \frac{\sum_{i=1}^{n} w'_{st}(x_s(V_i)) h(x_s(V_i))}{\sum_{i=1}^{n} w'_{st}(x_s(V_i))} \tag{7}$$

where $w'_{st}(x)$ are defined in the same way as the weights $w_{st}(x)$ but only up to a normalisation constant of the densities involved.

A special case arises when $s = t$. In that case, we have standard sampling:

$$\hat{\theta}_\Omega(t) = \hat{\theta}_{\Omega,t}(t) = \frac{1}{n}\sum_{i=1}^{n} h(x_t(V_i)) \tag{8}$$

because $w_{tt}(x) = 1$ for all x. This would lead to the following estimator for θ_* [8]:

$$\hat{\theta}_{*\Omega} = \hat{\theta}_\Omega(T_{*\Omega}) = \min_{t\in\mathscr{T}} \hat{\theta}_\Omega(t) \tag{9}$$

where

$$T_{*\Omega} = \arg\min_{t\in\mathscr{T}} \hat{\theta}_\Omega(t). \tag{10}$$

A difficulty with calculating $T_{*\Omega}$ is that we need to evaluate h at all sample points $x_t(V_i)$, and these points will be transformed as we change t in the optimizing algorithm which needs re-evaluation of h. With importance sampling, however, for fixed s, we only need to evaluate h for the sample points $x_s(V_i)$, independently of t. So if h is expensive to evaluate (e.g. finite element computations in engineering problems), then importance sampling is particularly useful, because we do not need to re-evaluate h for different t when optimizing over t.

For illustrating iterative importance sampling later on we introduce an estimator which depends on the importance sampling parameter s and on parameter t in which direction we are optimizing:

$$\hat{\vartheta}_\Omega(s,t) = \hat{\theta}_{\Omega,s}(t). \tag{11}$$

We note that this is an estimator of the function

$$\vartheta(s,t) = \theta(t) = \int h(x) f_t(x)\,\mathrm{d}x \tag{12}$$

which does not depend on s. The function $\hat{\vartheta}_\Omega$ has the following properties:

- In t-direction we have importance sampling for each fixed $s \in \mathscr{T}$.
- On the diagonal $s = t$ we have standard sampling since $\hat{\vartheta}_\Omega(s,s) = \hat{\theta}_\Omega(s)$.

By means of importance sampling we have the following estimator for θ_* depending on the importance sampling parameter s [8]:

$$\hat{\theta}_{*\Omega}(s) = \hat{\vartheta}_\Omega(s, \tau_{*\Omega}(s)) = \min_{t\in\mathscr{T}} \hat{\theta}_{\Omega,s}(t), \tag{13}$$

where

$$\tau_{*\Omega}(s) = \arg\min_{t\in\mathscr{T}} \hat{\vartheta}_\Omega(s,t) = \arg\min_{t\in\mathscr{T}} \hat{\theta}_{\Omega,s}(t). \tag{14}$$

Remark: Standard sampling means that the sample points have already the desired distribution with parameter t while importance sampling means in this context that the sample points are distributed according to another distribution with parameter s which has to be compensated by reweighting the sample as in Eq. (6). The original meaning of importance sampling is to put the sample points (choice of s) where it is "important" to reduce the variance of the estimator and then reweight as in Eq. (6).

Here, importance sampling is only an approximation of standard sampling to speed up the computations by reweighting the sample points instead of re-evaluating the function h. But there is no special choice for the importance sampling parameter s as in the original meaning. In [2] "real" importance sampling (design point method) was used to obtain upper probabilities of failure for a reliability engineering problem.

3. Iterative Importance Sampling

An issue with the importance sampling estimates is that their quality can be very poor if $\tau_{*\Omega}(s)$ is far from s, that means far from the diagonal $s = t$ where we have standard sampling. A procedure for iteratively improving the choice of s was proposed in [2, 7] and further developed and formalized in [9]. The procedure iteratively applies the operator $\tau_{*\Omega}$. Under the assumption that this iterative application

$$s^{(k+1)} = \tau_{*\Omega}(s^{(k)}), \quad k = 1,2,\ldots \tag{15}$$

of the operator $\tau_{*\Omega}$ reaches a unique fixed point, say $S_{*\Omega}$, our improved lower estimator is:

$$\hat{\theta}^{\dagger}_{*\Omega} = \hat{\vartheta}_\Omega(S_{*\Omega}, S_{*\Omega}) = \hat{\theta}_{\Omega,S_{*\Omega}}(S_{*\Omega}) = \frac{1}{n}\sum_{i=1}^{n} h(x_{S_{*\Omega}}(V_i)). \tag{16}$$

Since the fixed point $S_{*\Omega}$ is on the diagonal $s = t$ we have standard sampling and not only an approximation using importance sampling.

In numerical examples discussed in [2, 7, 8], normally, a fixed point is indeed obtained after few steps which is crucial to have an advantage over standard sampling. Needing too many iteration steps may be more expensive than using standard sampling as in Equation (9). However, $\tau_{*\Omega}$ is not necessarily continuous, and therefore it is not guaranteed that a fixed point exists. Even if it is continuous, $\tau_{*\Omega}$ is not necessarily contracting, and therefore it is not guaranteed that a fixed point can be found. In [9] an example was presented which shows that all three possible cases may happen depending on the sample size n:

- Divergence (circling) for $n = 1000$.
- No unique fixed point and converging to the "wrong" fixed point for $n = 10000$.
- Convergence to the unique fixed point for $n = 100000$.

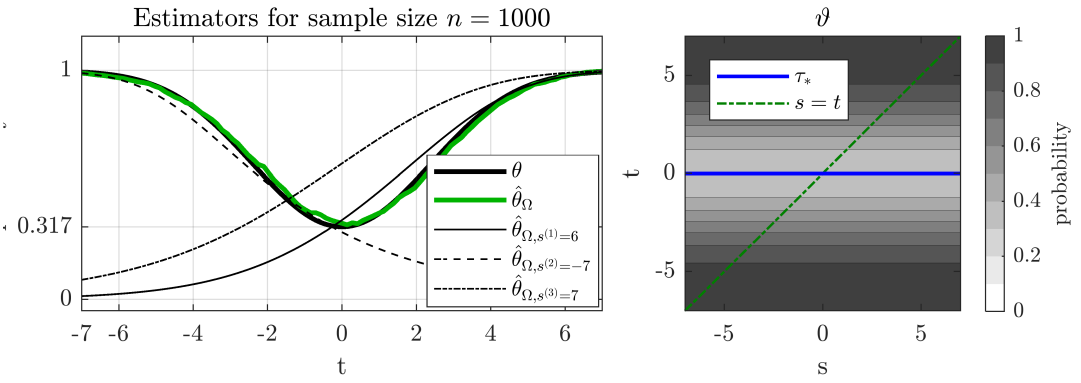


Figure 1: Exact probability θ, standard Monte Carlo estimation $\hat{\theta}_\Omega$ and importance sampling estimators $\hat{\theta}_{\Omega,s^{(1)}=6}$, $\hat{\theta}_{\Omega,s^{(2)}=-7}$, $\hat{\theta}_{\Omega,s^{(3)}=7}$ for the first three iteration steps and sample size $n = 1000$ (diverging case), as a function of $t \in \mathscr{T} = [-7,7]$ (left). Contour plot of exact function $\vartheta(s,t) = \theta(t)$ and its minimum $\tau_*(s) = t_* = 0$ (right), for comparison with the contour plots of the estimators in Figures 2 and 5.

Asymptotically, as the sample size n increases, $\tau_{*\Omega}$ should have a fixed point $S_{*\Omega}$, and both $T_{*\Omega}$ (from standard sampling) and $S_{*\Omega}$ should converge, in probability, to

$$t_* = \arg\min_t \theta(t). \tag{17}$$

The intuition behind this is that $\hat{\vartheta}_\Omega$ converges in probability to ϑ as the sample size goes to infinity. This also means that $\tau_{*\Omega}$ converges to

$$\tau_*(s) = \arg\min_{t\in\mathscr{T}} \vartheta(s,t). \tag{18}$$

But τ_* is constant in s and has a unique fixed point.

Remarks:

- If we already start the iteration at the fixed point $S_{*\Omega}$ (if existing) then we clearly have $S_{*\Omega} = \tau_{*\Omega}(S_{*\Omega})$ but only if we use always the same set Ω of basic random numbers, otherwise we would jump out of the fixed point.
- Since $\tau_{*\Omega}$ is not continuous in general, it may happen that the iteration switches between two values which lie very close to each other; decreased distance of the two values for increased sample size n.
- Let s be fixed. It is stated in [8], that the estimator $\hat{\theta}_{*\Omega}(s)$ is negatively biased and that the bias can be very large in specifically constructed examples. Furthermore there it is stated that $\hat{\theta}_{*\Omega}(s)$ is consistent if $\{h(x_t(\cdot)) : t \in \mathscr{T}\}$ is a Glivenko–Cantelli class.

Example: We consider the estimation of the lower probability of the event $D = (-\infty,-2] \cup [2,\infty)$ with respect to the set of normal distributions with mean $t \in \mathscr{T} = [-7,7]$ and variance $\sigma^2 = 4$. In this case the function h is the indicator function of the set D. For importance sampling we always use normalised weights.

The probability θ is depicted in Figure 1 (left) for each value $t \in \mathscr{T}$. The minimum (lower probability) θ_* is equal to 0.3173 and is achieved for $t_* = 0$. In addition, the estimators $\hat{\theta}_\Omega$ and $\hat{\theta}_{\Omega,s^{(i)}}$, $i = 1,2,3$, are depicted for $n = 1000$. In Figure 1 (right) the exact function ϑ is depicted as contour plot together with τ_* for comparison with estimators $\hat{\vartheta}_\Omega$ and $\tau_{*\Omega}$ later on. In Figure 2 we show the contour plots of $\hat{\vartheta}_\Omega$ for sample sizes $n = 1000, 10000, 100000$ and the iteration paths for starting point $s^{(1)} = 6$. For this example we can do that because h is very cheap to evaluate. In the first two cases there is no convergence because we are starting too far from the unique fixed point. In the third case, for a large sample size $n = 100000$, we achieve convergence.

An iteration path for a fixed point iteration is constructed in the following way: We start at $s = s^{(1)}$, go vertically up to the function value $\tau_{*\Omega}(s^{(1)})$ (blue function in the figures), then go horizontally to the diagonal $s = t$ (green line) and then again vertically up to $\tau_{*\Omega}(s^{(2)})$ and so on.

4. Combining Previous Results of Iteration to Improve Convergence

Let us look at the left contour plot of $\hat{\vartheta}_\Omega$ in Figure 2. There we start our iteration with $s^{(1)} = 6$ and get $\hat{\theta}_{\Omega,s^{(1)}=6}$. This approximation is quite good near 6 but very bad far from 6 which leads to a completely wrong minimum at -7 and let us jump to $s^{(2)} = -7$. Then, we have the opposite problem with $\hat{\theta}_{\Omega,s^{(1)}=-7}$ which leads to a wrong minimum at 7 and starts circling around. The idea is now to take into account that ϑ is constant in s and that we have already a good

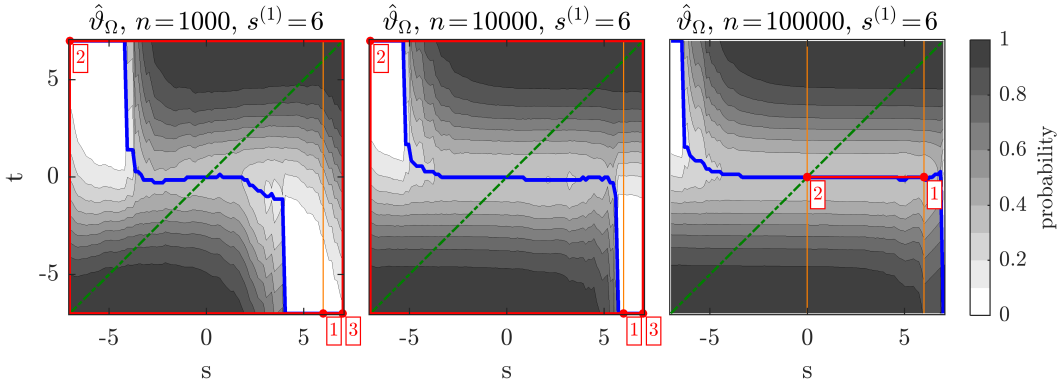


Figure 2: Contour plots of $\hat{\vartheta}_\Omega(s,t) = \hat{\theta}_{\Omega,s}(t)$ and depiction of $\tau_{*\Omega}(s) = \arg\min_{t\in\mathscr{T}} \hat{\theta}_{\Omega,s}(t)$ (blue line) for three different sample sizes $n = 1000$, 10000, and 100000. The path of the iteration with starting value $s^{(1)} = 6$ is plotted as a red line. We have divergence (circling) in the first two cases, due to bad approximation in regions where t is far from s. For the large sample size $n = 100000$ we have convergence.

approximation around 6 in the previous step. That means we have to combine both $\hat{\theta}_{\Omega,s^{(1)}=6}$ and $\hat{\theta}_{\Omega,s^{(2)}=-7}$ and to update $\tau_{*\Omega}$ to continue the iteration:

$$s^{(k+1)} = \tau_{*\Omega}^{(k)}\left(s^{(k)}, s^{(k-1)}, \ldots, s^{(1)}\right),\ k = 1,2,\ldots \tag{19}$$

We define this new iteration operator in the following way:

$$\tau_{*\Omega}^{(k)}\left(s^{(k)}, s^{(k-1)}, \ldots, s^{(1)}\right) = \arg\min_{t\in\mathscr{T}} \sum_{i=1}^{k} \varphi_{s^{(i)}}^{(k)}(t) \cdot \hat{\theta}_{\Omega,s^{(i)}}(t) \tag{20}$$

where a convex sum with the previous estimators is used.

The weighting functions $\varphi_{s^{(i)}}^{(k)}$ should have the following properties:

- Normalisation: $\sum_{i=1}^{k} \varphi_{s^{(i)}}^{(k)}(t) = 1$ for all $t \in \mathscr{T}$.
- High weights $\varphi_{s^{(i)}}^{(k)}(t)$ around $s^{(i)}$ where we have a good approximation and low weights $\varphi_{s^{(i)}}^{(k)}(t)$ far from $s^{(i)}$.

We will develop three different methods to obtain the weight functions. For some methods it may be easier to focus on the second property, obtaining weights $\tilde{\varphi}_{s^{(i)}}^{(k)}$ and normalise afterwards:

$$\varphi_{s^{(i)}}^{(k)}(t) = \tilde{\varphi}_{s^{(i)}}^{(k)}(t) \Big/ \sum_{j=1}^{k} \tilde{\varphi}_{s^{(j)}}^{(k)}(t) \quad \text{for all } t \in \mathscr{T}. \tag{21}$$

For visualization purpose we also update the estimator $\hat{\vartheta}_\Omega$ in the following way:

$$\hat{\vartheta}_\Omega^{(k)}(s,t) = \varphi_s^{(k)}(t) \cdot \hat{\vartheta}_\Omega(s,t) + \sum_{i=1}^{k-1} \varphi_{s^{(i)}}^{(k)}(t) \cdot \hat{\theta}_{\Omega,s^{(i)}}(t). \tag{22}$$

In addition, we plot the iteration path in the examples presented. We note that then clearly the previous steps of the iteration path not really fit into the contour plots because they are obtained before we have this new updated information.

4.1. Combination Method using Exponential Functions

Our first approach is to use an exponential function with the property that $\tilde{\varphi}_{s^{(i)}}^{(k)}$ is 1 at $t = s^{(i)}$ and that $\tilde{\varphi}_{s^{(i)}}^{(k)} > 0$ is smaller the greater the distance from t to $s^{(i)}$ is:

$$\tilde{\varphi}_{s^{(i)}}^{(k)}(t) = \mathrm{e}^{-\|t-s^{(i)}\|_D^2}. \tag{23}$$

In the formula appears the norm

$$\|t-s^{(i)}\|_D^2 = (t-s^{(i)})^{\mathsf{T}} D^2 (t-s^{(i)}) \tag{24}$$

with diagonal matrix

$$D = \begin{bmatrix} d_1 & & & \\ & d_2 & & \\ & & \ddots & \\ & & & d_m \end{bmatrix} \tag{25}$$

to measure the distance between t and $s^{(i)}$. The matrix D scales the components $t_j - s_j^{(i)}$ by d_j. This we need because of the different magnitude of the m components of the parameter vector and for describing what is "near" to $s^{(i)}$. In the case where

$$\mathscr{T} = [\underline{t}_1, \bar{t}_1] \times \cdots \times [\underline{t}_m, \bar{t}_m] \tag{26}$$

one could define

$$d_j = \frac{c}{\bar{t}_j - \underline{t}_j} \tag{27}$$

with a constant $c > 0$.

Example continued: For our one-dimensional example with $\mathscr{T} = [\underline{t}, \bar{t}] = [-7, 7]$ we use

$$D = \frac{30}{\bar{t} - \underline{t}} = \frac{30}{14} = 2.14. \tag{28}$$

In Figure 3(*a*) the combination $\hat{\theta}^{(2)}_{\Omega,s^{(2)}}$ of the estimators $\hat{\theta}_{\Omega,s^{(1)}}$ and $\hat{\theta}_{\Omega,s^{(2)}}$ is illustrated for the second iteration step and in Figure 4(*a*) the combination $\hat{\theta}^{(3)}_{\Omega,s^{(3)}}$ for the third iteration step for sample size $n = 1000$. In Figure 5(*a*) we show contour plots $\hat{\vartheta}^{(i)}_{\Omega}$, $i = 1, 2, 3$, and iteration paths for the first three iteration steps.

4.2. Combination Method using Effective Sample Size

For our second approach we use the formula [6]

$$n_{\text{eff},s}(t) = \frac{(\sum_{k=1}^{n} w_{st}(x_s(V_k)))^2}{\sum_{k=1}^{n} w_{st}(x_s(V_k))^2} \tag{29}$$

for the effective sample size and define

$$\tilde{\varphi}^{(k)}_{s^{(i)}}(t) = \frac{1}{n} \cdot n_{\text{eff},s^{(i)}}(t). \tag{30}$$

The effective sample size is equal to the sample size n for $t = s^{(i)}$ (standard sampling) and becomes smaller and smaller for t far from $s^{(i)}$. It is one of the measures for the efficiency of the importance sampling method.

The advantage is that this approach is very cheap because the weights in the formula for $n_{\text{eff},s^{(i)}}$ are already computed and that there is no need to think about the values of additional parameters such as $d_1, \ldots, d_m$ in the previous method with the exponential function.

Example continued: In Figures 3(*b*) and 4(*b*) the combinations $\hat{\theta}^{(2)}_{\Omega,s^{(2)}}$ and $\hat{\theta}^{(3)}_{\Omega,s^{(3)}}$ are illustrated for the second and third iteration step, respectively. In Figure 5(*b*) we show again the contour plots $\hat{\vartheta}^{(i)}_{\Omega}$ and iteration paths for the first three iteration steps.

4.3. Piecewise Multilinear Interpolation

Here, the idea is to use shape or basis functions as in the finite element method. One of the properties of such functions is that they are one at the corresponding node (here $s^{(i)}$) and zero at all other nodes (here all $s^{(j)} \neq s^{(i)}$). This is what we want to have because $\varphi^{(k)}_{s^{(i)}}$ for corresponding node $s^{(i)}$ should not be disturbed by other weighting functions at $s^{(i)}$. A further important property is that these basis functions are zero outside their surrounding elements.

For piecewise linear interpolation in one dimension assuming that $\mathscr{T} = [\underline{t}, \bar{t}]$ we have to sort the $s^{(i)}$:

$$(s^{(i_1)}, \ldots, s^{(i_k)}) = \text{sort}(s^{(1)}, \ldots, s^{(k)}). \tag{31}$$

Then the intervals $[\underline{t}, s^{(i_1)}], [s^{(i_1)}, s^{(i_2)}], \ldots, [s^{(i_k)}, \bar{t}]$ are the elements and the $\varphi^{(k)}_{s^{(i_j)}}$ are basis functions corresponding to nodes $s^{(i_j)}$. Here, we can directly define the $\varphi^{(k)}_{s^{(i_j)}}$ because they are already normalised due to the definition of the basis functions which are given in the following way with special treatment of the functions associated with $s^{(i_1)}$ and $s^{(i_k)}$.

$$\varphi^{(k)}_{s^{(i_j)}}(t) = \begin{cases} \dfrac{t - s^{(i_{j-1})}}{s^{(i_j)} - s^{(i_{j-1})}} & t \in [s^{(i_{j-1})}, s^{(i_j)}],\ j = 2, \ldots, k, \\ \dfrac{s^{(i_{j+1})} - t}{s^{(i_{j+1})} - s^{(i_j)}} & t \in [s^{(i_j)}, s^{(i_{j+1})}],\ j = 1, \ldots, k-1, \\ 1 & t \in [\underline{t}, s^{(i_1)}],\ j = 1, \\ 1 & t \in [s^{(i_k)}, \bar{t}],\ j = k, \\ 0 & \text{otherwise.} \end{cases} \tag{32}$$

The functions in the two special cases are defined to be 1 from or to the corresponding boundary ($\underline{t}$ and $\bar{t}$).

In higher dimensions, $m > 1$, it is more difficult. In the case where the parameter set $\mathscr{T}$ is in the form of

$$\mathscr{T} = [\underline{t}_1, \bar{t}_1] \times \cdots \times [\underline{t}_m, \bar{t}_m] \tag{33}$$

the MATLAB command `delaunayn` can be used to get an m-dimensional triangulation of $\mathscr{T}$ with respect the points $s^{(1)}, \ldots, s^{(k)}$ together with all corners $c^{(i)}$ of $\mathscr{T}$. To obtain $\varphi^{(k)}_{s^{(i)}}$ by means of the MATLAB command `gridatan` we have to set the function values v at the points $s^{(1)}, \ldots, s^{(k)}$ and in the corners $c^{(j)}$ of $\mathscr{T}$:

$$v(s^{(j)}) = \begin{cases} 1 & j = i, \\ 0 & \text{otherwise} \end{cases} \tag{34}$$

and

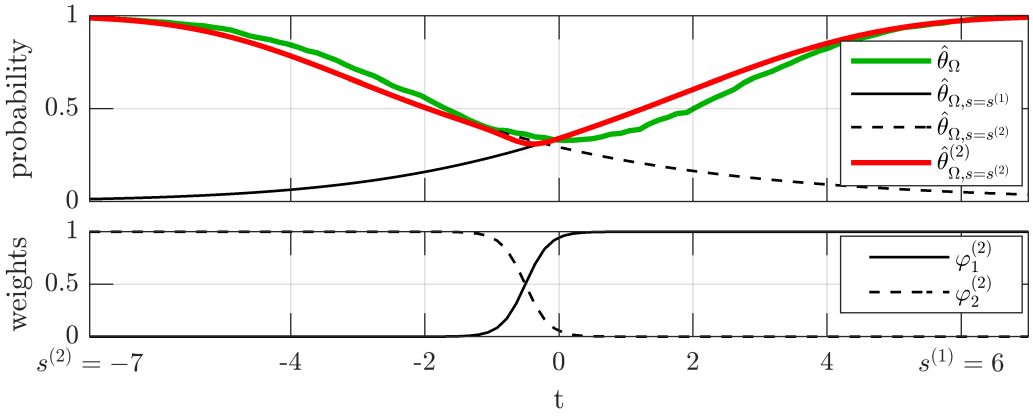
$$v(c^{(j)}) = \begin{cases} 1 & c^{(j)} \text{ and } s^{(i)} \text{ are in the same triangle,} \\ 0 & \text{otherwise.} \end{cases} \tag{35}$$

This corresponds to the special treatment of the first and the last $s^{(i_j)}$ in the one-dimensional case.

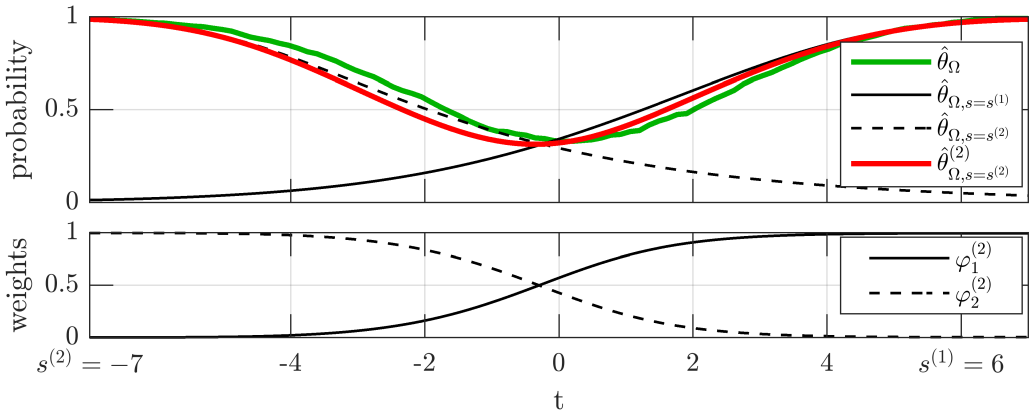
Example continued: The results are depicted in Figures 3(*c*), 4(*c*) and 5(*c*) in the same way as for the two other methods presented before.

Summary and Conclusion

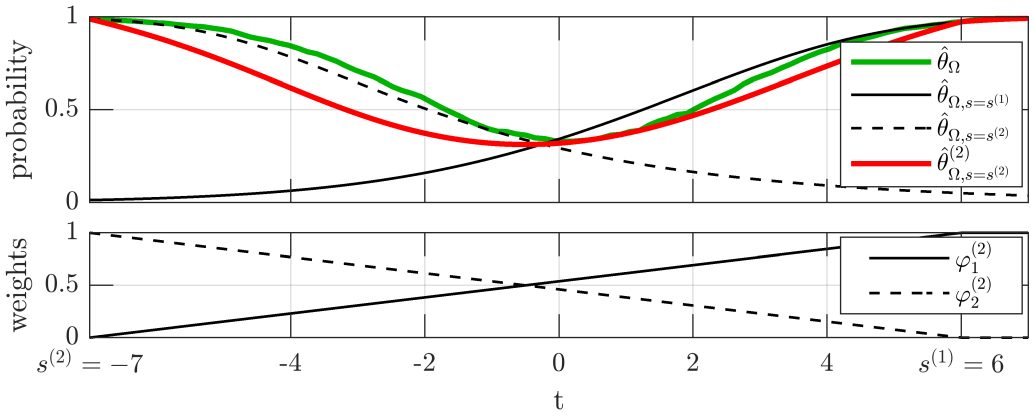
We extended the iterative importance sampling algorithm developed in [9] by new methods to improve the convergence. These methods are using information and results from previous iteration steps which are available in the algorithm anyway without expensive additional calculations. This allows us to use two different strategies (increasing sample coverage presented in [9] and combining estimators



(*a*) Combination method: exponential function.

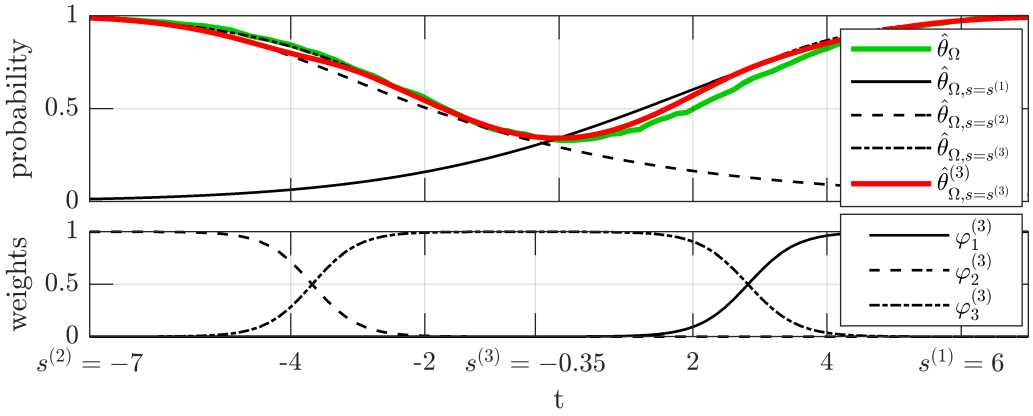


(*b*) Combination method: effective sample size.

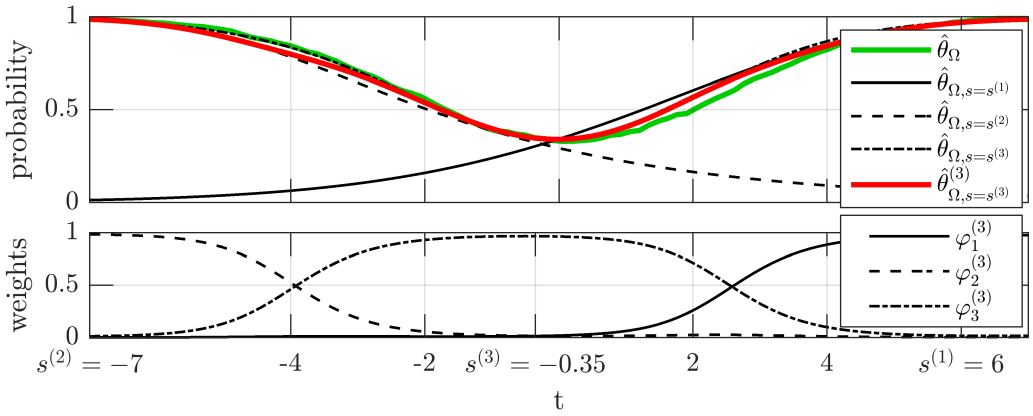


(*c*) Combination method: piecewise linear interpolation.

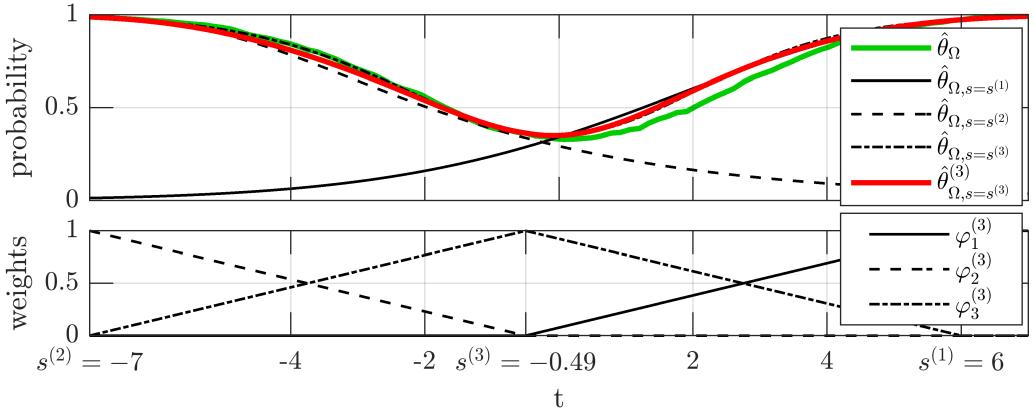
Figure 3: We illustrate the combination of $\hat{\theta}_{\Omega,s^{(1)}}$ and $\hat{\theta}_{\Omega,s^{(2)}}$ for the second iteration step in the iterative importance sampling algorithm for the following combination methods: (a) exponential function, (b) effective sample size and (c) piecewise linear interpolation. The functions $\hat{\theta}_\Omega$ (standard sampling), $\hat{\theta}_{\Omega,s^{(1)}}$, $\hat{\theta}_{\Omega,s^{(2)}}$ (importance sampling for iteration steps $s^{(1)} = 6$ and $s^{(2)} = -7$) and their combination $\hat{\theta}^{(2)}_{\Omega,s^{(2)}}$ for the second iteration step are depicted. In addition, the weight functions $\varphi^{(k)}_{s^{(i)}}$, $i = 1,2$, needed for the combination are plotted.



(*a*) Combination method: exponential function.

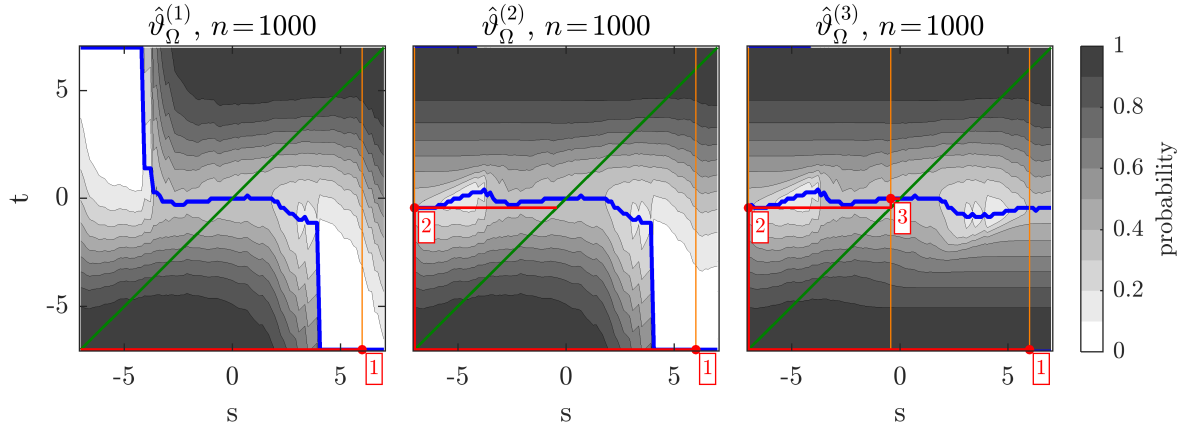


(*b*) Combination method: effective sample size.

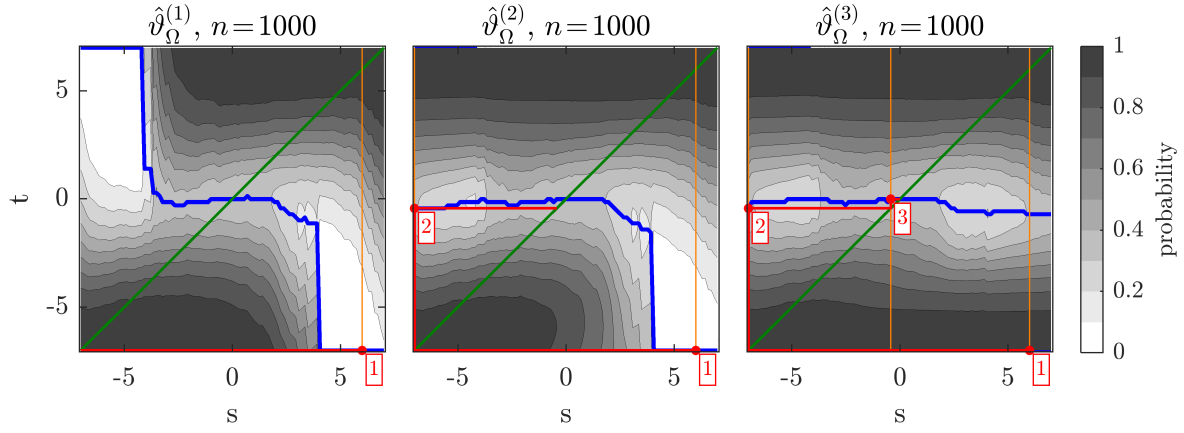


(*c*) Combination method: piecwise linear interpolation.

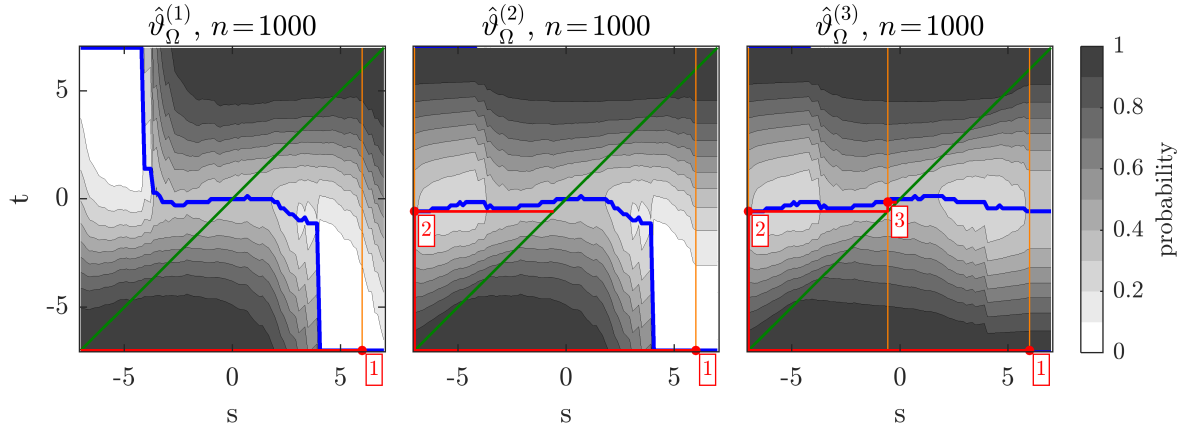
Figure 4: We illustrate the combination of $\hat{\theta}_{\Omega,s^{(1)}}$, $\hat{\theta}_{\Omega,s^{(2)}}$ and $\hat{\theta}_{\Omega,s^{(3)}}$ for the third iteration step in the iterative importance sampling algorithm for the following combination methods: (a) exponential function, (b) effective sample size and (c) piecewise linear interpolation. The functions $\hat{\theta}_\Omega$ (standard sampling), $\hat{\theta}_{\Omega,s^{(1)}}$, $\hat{\theta}_{\Omega,s^{(2)}}$, $\hat{\theta}_{\Omega,s^{(3)}}$ (importance sampling for iteration steps $s^{(1)} = 6$, $s^{(2)} = -7$ and $s^{(3)}$) and their combinations $\hat{\theta}^{(3)}_{\Omega,s^{(3)}}$ for the third iteration step are depicted. In addition, the weight functions $\varphi^{(k)}_{s^{(i)}}$, $i = 1,2,3$, needed for the combination are plotted.



(*a*) Combination method: exponential function.



(*b*) Combination method: effective sample size.



(*c*) Combination method: piecewise linear interpolation.

Figure 5: Contour plots of $\hat{\vartheta}_\Omega^{(i)}$ and depiction of $\tau_{*\Omega}^{(i)}$ (blue line) for three iteration steps, $i = 1,2,3$, for each of the different combination methods. The path of the iteration with starting value $s^{(1)} = 6$ is plotted as a red line. We have convergence in all three cases.

from previous iteration steps) for improving convergence in the iterative importance sampling algorithm at the same time. This is in particular useful if one of the two methods does not work well.

The approach with the effective sample size is cheap and convenient to apply and leads to good approximations already after two iteration steps. After three steps there is almost no difference between the methods comparing the new estimators $\hat{\theta}^{(3)}_{\Omega,s^{(3)}}$. Comparing the contour plots of $\hat{\vartheta}^{(3)}_{\Omega}$ we have fast convergence to the exact ϑ, in particular for the effective sampling size method and for the piecewise interpolation method.

References

[1] George E. P. Box and Mervin E. Muller. A note on the generation of random normal deviates. *The Annals of Mathematical Statistics*, 29(2):610–611, 1958.

[2] Thomas Fetz. Efficient computation of upper probabilities of failure. In Christian Bucher, Bruce R. Ellingwood, and Dan M. Frangopol, editors, *12th International Conference on Structural Safety & Reliability*, pages 493–502, 2017.

[3] Thomas Fetz and Michael Oberguggenberger. Imprecise random variables, random sets, and Monte Carlo simulation. In Thomas Augustin, Serena Doria, Enrique Miranda, and Erik Quaeghebeur, editors, *ISIPTA'15: Proceedings of the Ninth International Symposium on Imprecise Probability: Theories and Applications*, pages 137–146, 2015. URL http://www.sipta.org/isipta15/data/paper/12.pdf.

[4] Thomas Fetz and Michael Oberguggenberger. Imprecise random variables, random sets, and Monte Carlo simulation. *Int. J. of Approximate Reasoning*, 78:252–264, 2016.

[5] Serafín Moral and Nic Wilson. Importance sampling Monte-Carlo algorithms for the calculation of Dempster-Shafer belief. In *Proceeding of IPMU'96*, pages 1337–1344, 1996.

[6] Art B. Owen. *Monte Carlo theory, methods and examples*. 2013. URL http://statweb.stanford.edu/~owen/mc/.

[7] Matthias C. M. Troffaes. A note on imprecise Monte Carlo over credal sets via importance sampling. In Alessandro Antonucci, Giorgio Corani, Inés Couso, and Sébastien Destercke, editors, *Proceedings of the Tenth International Symposium on Imprecise Probability: Theories and Applications*, volume 62 of *Proceedings of Machine Learning Research*, pages 325–332. PMLR, July 2017. URL http://proceedings.mlr.press/v62/troffaes17a.html.

[8] Matthias C. M. Troffaes. Imprecise Monte Carlo simulation and iterative importance sampling for the estimation of lower previsions. *Int. J. of Approximate Reasoning*, 101:31–48, 2018.

[9] Matthias C. M. Troffaes, Thomas Fetz, and Michael Oberguggenberger. Iterative importance sampling for estimating expectation bounds under partial probability specifications. In Marco De Angelis, editor, *Proceedings of the 8th International Workshop on Reliable Engineering Computing*, pages 147–154, Liverpool, 2018. Liverpool University Press.

[10] H. Zhang. Interval importance sampling method for finite element-based structural reliability assessment under parameter uncertainties. *Structural Safety*, 38: 1–10, 2012.

[11] Jiaxin Zhang and Michael D. Shields. On the quantification and efficient propagation of imprecise probabilities resulting from small datasets. *Mechanical Systems and Signal Processing*, 98:465–483, 2018.