

Solving chance constrained optimal control problems in aerospace via kernel density estimation

J.-B. Caillau¹  | M. Cerf² | A. Sassi³ | E. Trélat⁴ | H. Zidani³

¹Université de Bourgogne Franche-Comté & CNRS/Inria, Dijon, France

²Ariane Group, Paris, France

³ENSTA-ParisTech, Paris, France

⁴Sorbonne Université & CNRS/Inria Paris, France

Correspondence

J.-B. Caillau, LJAD, Université Côte d'Azur & CNRS/Inria Parc Valrose, F-06108 Nice
Email: caillau@unice.fr

Funding information

7th Framework Programme Marie Curie Initial Training Network
'FP7-PEOPLE-2010-ITN', Grant/Award Number: 264735-SADCO; Institute iCODE, IDEX Paris-Saclay, Grant/Award Number: ANR-11-IDEX-0003-02

Summary

The goal of this paper is to show how nonparametric statistics can be used to solve some chance constrained optimization and optimal control problems. We use the kernel density estimation method to approximate the probability density function of a random variable with unknown distribution from a relatively small sample. We then show how this technique can be applied and implemented for a class of problems including the Goddard problem and the trajectory optimization of an Ariane five-like launcher.

KEYWORDS

aerospace engineering, chance constrained optimization, kernel density estimation, optimal control

1 | INTRODUCTION

This paper is dedicated to a numerical approach for solving chance constrained optimal control problems using the kernel density estimation (KDE) technique. One of the earliest and most famous examples of optimal control problems in aerospace dates back to the beginning of the twentieth century. In 1921, American physicist R. H. Goddard published a paper¹ in which he studied the problem of minimizing the fuel consumption of a rocket ascending vertically from Earth's surface, taking into account both atmospheric drag and gravitational field. In order to better explain the nature of this kind of problems, we will give a simplified model. Consider the vertical ascent of a rocket in one dimension. Function $r(t)$ represents the rocket altitude, $v(t)$ its speed, and $m(t)$ its mass. We introduce the variable $u(t) \in [0, 1]$, which defines the rate of the maximum thrust applied at a given time. The vehicle starts from a still position at ground level, at time $t = 0$ the thrust force $Tu(t)$ of the engine pushes the launcher upwards against the force of gravity $m(t)g$ with a fuel consumption rate $\frac{T}{v_e}u(t)$, where v_e is the fuel speed. The maximum thrust T , the fuel speed v_e , the initial mass m_0 , and the final time t_f are fixed. The controlled system describing the launcher's dynamics is

$$\begin{cases} \dot{r}(t) = v(t) & t \in [0, t_f], \\ \dot{v}(t) = \frac{T}{m(t)}u(t) - g & t \in [0, t_f], \\ \dot{m}(t) = -\frac{T}{v_e}u(t) & t \in [0, t_f], \\ (r(0), v(0), m(0)) = (0, 0, m_0), \end{cases} \quad (1)$$

where the admissible control set is

$$\mathcal{U} := \{u : \mathbf{R}_+ \rightarrow [0, 1] \subset \mathbf{R} \mid u \text{ is measurable}\}.$$

We want to solve the optimal control problem of finding a particular $u^* \in \mathcal{U}$ that maximizes the final mass of the launcher while ensuring that it reaches at least a given altitude ρ_f at time t_f . Formally, this translates to solving

$$\begin{cases} \max_{u \in \mathcal{U}} m_f(u, T), \\ r_f(u, T) \geq \rho_f, \end{cases} \quad (2)$$

where $m_f(u, T)$ and $r_f(u, T)$ are the final mass and altitude associated to $u \in \mathcal{U}$ and a choice of parameter T by (1). For a general theoretical study of this kind of problems, we refer to the works of Agrachev and Sachkov² and Vinter.³ Robust methods are aimed at achieving consistent performance and/or stability in the presence of bounded modeling errors. Drawing a parallel with the example, let us suppose that the thrust T is estimated with some margin errors and assume that we want to maximize the final payload in the presence of these uncertainties on T . One possible approach consists in the use of worst-case analysis to treat uncertainties to obtain what is called a “robust” solution. Using a variation of Wald’s maximin model,⁴ one can consider the so-called *robust optimal control problem*

$$\begin{cases} \max_{u \in \mathcal{U}} \min_{T \in [T_-, T_+]} m_f(u, T), \\ r_f(u, T) \geq \rho_f, \quad T \in [T_-, T_+], \end{cases} \quad (3)$$

with the same notations as before for $m_f(u, T)$ and $r_f(u, T)$, associated to (u, T) by (1). A solution to this problem would be a control strategy $u^* \in \mathcal{U}$ that maximizes the final mass of the launcher even for the worst realization of the parameter T while satisfying the constraint for the final altitude, for any $T \in [T_-, T_+]$. As pointed out in the work of Berstimas and Sim,⁵ robust optimization requires a trade-off. The price to obtain a solution that is feasible in every scenario often results in the suboptimality of the value function. Moreover, there might exist problems in which the constraint function cannot be satisfied for every realization of the model’s parameters.

Another approach used for solving robust optimization problems consists in chance constrained optimization. The name comes from the idea of treating the uncertainties in the underlying mathematical model as random variables. More precisely, in the case of our example, we assume that T is a random variable taking values inside an interval $[T_-, T_+]$, according to a given probability distribution. As a consequence of this definition, the two functions $m_f(u, T)$ and $r_f(u, T)$ also become random variables. We introduce the parameter $p \in [0, 1]$ and consider the following problem:

$$\begin{cases} \max_{u \in \mathcal{U}} E[m_f(u, T)], \\ P(r_f(u, T) \geq \rho_f) \geq p, \end{cases}$$

where E denotes the expectation and P the probability. Here, p acts as a probability threshold for the realization of the event $r_f(u, T) \geq \rho_f$, and the inequality $P(r_f(u, T) \geq \rho_f) \geq p$ is called *chance* or *probability* constraint. Optimal control problems with chance constraints are often considered if there is a need for minimizing a cost associated to the performance of a dynamical model while taking into account uncertainties in the parameters defining it (see also the works of Serra et al^{6,7} for other computations of probabilities in a dynamical setting related to aerospace engineering.) In this paper, we study the numerical solution to chance constrained optimal control problems of the form

$$\begin{cases} \min_{(y, u) \in \mathcal{Y} \times \mathcal{U}} J(y, u), \\ P(G(y, u, \xi) \geq 0) \geq p, \end{cases}$$

where functions J and G may depend on both a finite number of optimization variables y and on a control function u . Note that we do not consider state constraints but only pure control constraints (these are included in the definition of the admissible set of controls, ie, $\mathcal{U}$) and endpoint constraints. As will be clear for the examples treated in Section 4, we assume that the expectation that should define the cost can be analytically computed so that the performance index J only depends on the deterministic decision and control variables. The parameter $p \in (0, 1)$ is a probability threshold and ξ is an m -dimensional random vector defined on a given probability space. We explore the application of the KDE. This

technique is used in nonparametric statistics to approximate the probability density function (PDF) of a random variable with unknown distribution. The main difficulty lies in the form of the constraint function. G being dependent on both y , u , and ξ , it is not a trivial task to derive an analytical representation of its probability distribution even if the distribution of ξ is known. The idea of applying nonparametric density estimation (and in particular KDE) to chance constrained optimization problems is not new (see for instance the works of Sahin and Diwekar⁸ and Calfa et al.,⁹ where this technique has been applied to an optimization problem in finite dimension). For the same type of problem, one can also mention the work of Nemirovski and Shapiro,¹⁰ where the authors use a different technique called the *scenario approach*. The work of Zhang et al.¹¹ features the use of KDE for solving problems in the simpler case where the optimization variables y are separated from the random variables ξ . To the best of our knowledge, however, KDE has not been used previously as a tool for solving optimal control problems, where both the cost and the constraints explicitly depend on a control function u .

The goal of this paper is to show the relevance of the KDE approach on some optimal control problems. This method can lead to very good results, hopefully inspiring new developments in the field of chance constrained optimal control. In Section 2, we present some existing results on the subject of chance constrained optimization. Section 3 gives an overview of the KDE technique and introduces the algorithm that we propose to use for solving chance constrained control problems. Section 4 consists of three numerical examples involving the application of KDE to chance constrained optimization and optimal control problems.

2 | CHANCE CONSTRAINED OPTIMIZATIONS: A BRIEF SURVEY

There exists a wide literature on the subject of chance constrained optimization, in particular, in the case of problems involving only an array of decision variables y in $\mathcal{Y} \subset \mathbf{R}^n$. As already mentioned, the robust approach to parametric optimization comes with some disadvantages. It might be difficult to guarantee the existence of a solution due to the strictness of the constraints, and even in the case of a relaxation approach, it might be hard to make sure that the problem satisfies all the required controllability hypotheses. Suppose that ξ is an m -dimensional random vector defined on some probability space $(\Omega, \mathcal{A}, P)$. Consider then the chance constrained problem

$$\begin{cases} \min_{y \in \mathcal{Y}} E[J(y, \xi)], \\ P(G(y, \xi) \geq 0) \geq p, \end{cases} \quad (4)$$

where $\mathcal{Y} \subset \mathbf{R}^n$ is the admissible set for the decision variables y , $J : \mathcal{Y} \times \mathbf{R}^m \rightarrow \mathbf{R}$ is an objective, $G : \mathcal{Y} \times \mathbf{R}^m \rightarrow \mathbf{R}$ defines a constraint inequality, $p \in (0, 1)$ is a probability threshold called *confidence level*. This kind of problem has been treated since the fifties.¹² A general theory is due to the work of Prékopa,^{13,14} who also introduced the convexity theory based on logconcavity. Other contributions on the logconcavity theory in stochastic programming can be found in related works.¹⁵⁻¹⁷ There exist many results on the regularity of the constraint function and on the error between approximated solutions of chance constrained optimization problems. Two fundamental theorems regarding continuity and convexity of the constraint function have been proven in the works of Prékopa¹⁴ and Raik.¹⁸ The continuity theorem has proven in the work of Raik¹⁸ that, if the function $G(\cdot, \xi)$ is upper semicontinuous (respectively continuous), then the function

$$\Gamma(y) := P(G(y, \xi) \geq 0)$$

is upper semicontinuous (respectively continuous). Additionally, the convexity theorem in the work of Prékopa¹⁴ states that, if G is quasi-concave and ξ has a log-concave probability distribution, then Γ is log-concave. Let us mention that the results presented in the works of Prékopa¹⁴ and Raik¹⁸ apply to the more general case of multiple joint constraints. In the work of van Ackooij and Henrion,¹⁹ the authors prove that, if the random array ξ has a Gaussian distribution, it is possible to obtain a gradient formula for the nonlinear probabilistic constraint Γ . The main feature of this result is that it opens the path to many numerical approaches based on descent algorithms. Moreover, obtaining the gradient of the chance constraint is a crucial step toward establishing first-order necessary conditions for optimality (see also the works of Uryasev²⁰ and Marti²¹). When the probability distribution of ξ is not precisely known and is replaced by an estimator, the result proven in the work of Henrion et al.²² gives the hypotheses allowing to estimate the difference between the solution of the original problem and the one where the estimator has been used.

In the literature, chance constrained optimal control problems have also been treated with other techniques, such as the scenario approach previously mentioned.²³⁻²⁵ A strong asset of this method is to provide a priori certificates for the chance

constraint. One difficulty to apply it in our optimal control context is that, for some samples, the problem might not be feasible (lack of controllability). As we will see further, this is not an issue for a KDE-based approach. Another technique to deal with chance constraints is the so-called back-mapping approach.^{26,27} The method is appealing because it avoids computing the probability distribution of the random variable involved in the chance constraint. One drawback, though, is that it requires some monotonic dependence with respect to some of the uncertain variables. For the applications in aerospace engineering we want to tackle, we do not want to rely on such an assumption. Among alternative techniques for solving chance constrained problems, one must cite Monte Carlo algorithms. Such an algorithm consists in repeatedly sampling variables and parameters of a problem to obtain numerical results, treating them as random quantities. This kind of approach might be very useful in the case of problems involving a high number of dimensions, many degrees of freedom, or unknown probability distributions. The main step of a method belonging to the Monte Carlo class consists in choosing a probability distribution for the inputs and generating random input values over the domain. The mathematical theory supporting these methods depends on the particular type chosen, but the main result on which Monte Carlo methods lay foundation is the strong law of large numbers. This class of methods possesses many advantages. They are usually easy to implement and can be easily parallelized if the random variables to be sampled are independent. Moreover, given the wide variety of existing Monte Carlo methods, it is not difficult to find an implementation specifically designed for a particular field. An important technique rising from the combination of the Monte Carlo method with the iterative gradient method is the stochastic Arrow-Hurwicz algorithm (SAHA) (see the work of Andrieu et al²⁸). This algorithm can be used to solve an optimization problem in the form

$$\begin{cases} \min_{y \in \mathcal{Y}} E[J(y, \xi)], \\ E[H(y, \xi)] \leq \alpha, \end{cases} \quad (5)$$

where the function H has to be regular enough to guarantee the convexity and the connectedness of the feasible subset defined by the constraint. Unfortunately, in the case of chance constrained optimization, these regularity properties of the feasible are not easily checked. Moreover, reformulating the chance constraint as a constraint on expectation requires some careful regularization process to be able to evaluate gradients of the constraint. See the work of Andrieu et al²⁸ for a more detailed explanation of SAHA and of the aforementioned difficulties.

The approach we present in the next section for solving numerically the chance constrained control problems is based on KDE. This technique consists in approximating the PDF of a random variable with an unknown distribution from a given sample. Contrary to Monte Carlo-based methods that often require a large number of simulations, in practice, KDE is able to provide good approximations of densities on the basis of a limited number of samples. In our chance constrained optimal control setting, KDE also has the advantage over the SAHA that no fine tuning of parameters is required to obtain a satisfactory estimation of the solution.

3 | KDE FOR CHANCE CONSTRAINED OPTIMIZATION PROBLEMS

3.1 | An overview of KDE techniques

Let X be a random variable with an unknown distribution f that we want to estimate and let $\{x_1, x_2, \dots, x_n\}$ be a sample of size n of the variable X . A KDE for the PDF f is the function

$$\hat{f}_{n,h}(x) := \frac{1}{nh} \sum_{i=1}^n K\left(\frac{x - x_i}{h}\right), \quad (6)$$

where the function K is the *kernel* and the smoothing parameter h the *bandwidth*. The earlier mentions of this method in its current form date back to the early 50s in the works of Rosenblatt²⁹ and Parzen,³⁰ this is why it is also known as the *Parzen-Rosenblatt window*. Silverman's book³¹ represents the basic text on the subject, whereas the work of Terrell and Scott³² provides a detailed analysis on the various properties of this technique. This method has also been applied to many other fields like archaeology, banking, climatology, economics, genetics, hydrology, and physiology (see the work of Sheather³³ for more references). A fundamental consistency result was obtained by Nadaraya.³⁴ Variations of this theorem have been studied in the works of Silverman³⁵ and Devroye and Györfi,³⁶ whereas an earlier but less general version of this result can also be found in the work of Parzen.³⁰

Theorem 1. (See the work of Nadaraya³⁴)

If the kernel $K : \mathbf{R} \rightarrow \mathbf{R}_+$ is a function of bounded variation, $f : \mathbf{R} \rightarrow \mathbf{R}_+$ is a uniformly continuous density function and, if h satisfies $\sum_{n=1}^{\infty} e^{-\gamma nh^2} < \infty$ for all positive γ , then

$$P \left(\lim_{n \rightarrow \infty} \sup_x |\hat{f}_{n,h}(x) - f(x)| = 0 \right) = 1.$$

The approximation error between f and $\hat{f}_{n,h}$ depends on the choice of both K and h . The kernel K is generally chosen such that it satisfies the conditions

$$\int K(x) dx = 1 \quad \text{and} \quad \int xK(x) dx = 0 \quad \text{and} \quad \int x^2 K(x) dx > 0.$$

For the sake of completeness, more details on kernel and bandwidth selection are included in Appendix A.

3.2 | Application of KDE approach to chance constrained optimization

Consider the following problem:

$$\begin{cases} \min_{(y,u) \in \mathcal{Y} \times \mathcal{U}} J(y,u), \\ P(G(y,u,\xi) \geq 0) \geq p, \end{cases} \quad (7)$$

where $\mathcal{Y} \subset \mathbf{R}^d$ and $\mathcal{U}$ are respectively the admissible sets for the decision variables y and the control u , and $J : \mathcal{Y}^d \times \mathcal{U} \rightarrow \mathbf{R}$ and $G : \mathcal{Y}^d \times \mathcal{U} \times \mathbf{R}^m \rightarrow \mathbf{R}$ are respectively the cost and constraint functions. Depending on the context, J and G might only be defined on a proper subset of $\mathcal{Y} \times \mathcal{U}$. By using KDE, we are able to produce an approximation of the PDF, defining the chance constraint, thus allowing us to replace the probability with the integral of the estimated PDF and solve the stochastic optimization problem as a deterministic one. For given y in $\mathcal{Y}$ and u in $\mathcal{U}$, let $f_{y,u}$ and $\hat{f}_{y,u}$ denote respectively the PDF of random variable $G(y,u,\xi(\cdot))$ (which is parameterized by y and u) and its approximation. To keep notation as clear as possible, we drop the subscripts n and h used in (6). Still, one has to remember that the approximated PDF, ie, $\hat{f}_{y,u}$, actually depend on the number of samples, on the kernel, and on the bandwidth. One has

$$P(G(y,u,\xi) \geq 0) = 1 - P(G(y,u,\xi) < 0) = 1 - \int_{-\infty}^0 f_{y,u}(x) dx = 1 - F_{y,u}(0),$$

where, for a given $(y,u) \in \mathcal{Y} \times \mathcal{U}$, $F_{y,u}$ denotes the cumulated density function (CDF) of the random variable $G(y,u,\xi(\cdot))$. We then build the estimator $\hat{f}_{y,u}$ of $f_{y,u}$ via KDE. By defining $\hat{F}_{y,u}(s) := \int_{-\infty}^s \hat{f}_{y,u}(x) dx$, we can write an approximation of our chance constraint in the form

$$\begin{cases} \min_{(y,u) \in \mathcal{Y} \times \mathcal{U}} J(y,u), \\ \hat{F}_{y,u}(0) \leq 1 - p. \end{cases} \quad (8)$$

Let (y^*, u^*) and $(\hat{y}^*, \hat{u}^*)$ be respectively the solutions of problems (7) and (8). Even in the absence of an explicit estimate for the error between (y^*, u^*) and $(\hat{y}^*, \hat{u}^*)$ by means of the error between $\hat{f}_{\hat{y}^*, \hat{u}^*}$ and f_{y^*, u^*} , we can always rely on the law of large numbers for the validation of our results a posteriori. The numerical solution of the chance constrained problem goes along the following steps.

- i. **Draw the sample.** Take a sample $\{\xi_1, \xi_2, \dots, \xi_n\}$ of size n from the random vector ξ . This operation has to be done only once at the beginning of the optimization procedure because the realizations of ξ are only dependent on its distribution and not on the decision variables and control y and u . For any value of (y,u) , such a sample defines n realizations $G(y,u,\xi_1), \dots, G(y,u,\xi_n)$ of the random variable $G(y,u,\xi(\cdot))$. In the numerical simulations presented in Section 4, for each size of sample, we draw several samples (typically ten simulations) and average. With an exception for Example 2 where, for the scalar and uniformly distributed random parameter, a single uniform sample is used.
- ii. **Compute the constraint function.** We need to define the value of $\hat{F}_{y,u}$ for any value of y and u . Given $(y,u) \in \mathcal{Y} \times \mathcal{U}$, for each element ξ_i of the sample, the constraint function $G(y,u,\xi_i)$ is evaluated. This allows to compute the approximate of the PDF according to (6) on the basis of a choice of kernel and bandwidth. For the numerical

computations of Section 4, we use a Gaussian kernel with the SNR bandwidth documented in Appendix A. In order to estimate

$$\hat{F}_{y,u}(s) := \int_{-\infty}^s \hat{f}_{y,u}(x) dx,$$

a quadrature rule is used to estimate the integral of the approximated density function. This implies a two-level scheme of approximation of the chance constraint, ie, KDE for the PDF estimation, then a quadrature to approximate the CDF of the estimated density. Regarding the quadrature, we employ the composite Simpson rule. Given an interval $[a, b]$, the integral of the function f is computed by dividing $[a, b]$ into an even number N of subintervals (in our case, $N = 2000$) and applying the formula

$$\int_a^b f(x) dx \simeq \frac{1}{3} \frac{b-a}{N} \left(f(a) + 2 \sum_{i=1}^{\frac{N}{2}-1} f(x_{2i}) + 4 \sum_{i=1}^{\frac{N}{2}} f(x_{2i-1}) + f(b) \right),$$

where $x_i := a + \frac{b-a}{N}i$, $i \in \{0, 1, \dots, N-1\}$. We decided to use Simpson's rule because it provides a good balance between the ease of code implementation and precision because the error of this quadrature formula is bounded by $\left(\frac{b-a}{N}\right)^4 (b-a) \max_{x \in [a,b]} |f^{(4)}(x)|$. Details and results on this formula can be found in the work of Young and Gregory.³⁷

Note that these computations have to be repeated every time we need to evaluate the constraint involving $\hat{F}_{y,u}$ at some (y, u) . As this entails evaluating through KDE (involving a large enough number n of samples) the PDF inside the quadrature, this step is the main bottleneck of the proposed method.

- iii. Solve the approximated problem.** Now that the approximation $\hat{F}_{y,u}$ of $F_{y,u}$ has been defined, we can solve problem (8) as a regular deterministic optimization problem. The numerical results in the next section have been obtained by using Fortran 90 to write the code interface and WORHP* as an SQP solver. Some preliminary tests were also made with IPOPT[†] but gave less satisfactory results. This solver is designed to handle finite dimensional nonlinear optimization problem in the form

$$\begin{cases} \min_{X \in \mathbf{R}^N} F(X), \\ X_L \leq X \leq X_U, \\ G_L \leq G(X) \leq G_U, \end{cases} \quad (9)$$

where $N \in \mathbf{N}$ is the number of decision (or optimization) variables, which are collected in the array $X := (X_1, X_2, \dots, X_N)$; the function $F(X) : \mathbf{R}^N \rightarrow \mathbf{R}$ represents the cost to be minimized; and $G(X) : \mathbf{R}^N \rightarrow \mathbf{R}^M$ is the constraint function, with $M \in \mathbf{N}$ being the number of constraints to be satisfied. The arrays $X_L, X_U \in \mathbf{R}^N$ and $G_L, G_U \in \mathbf{R}^M$ define respectively the lower and upper bounds for X and G . In addition to this, the user must provide an initial guess X_0 for the solution of (9), whereas the derivatives of F and G are optional because they can be approximated by the solver. We used this option so that gradients are computed using finite differences. A refinement would be to use automatic differentiation, which is possible whenever the dynamics involved in the examples are integrated by means of numerical schemes with a fixed grid (see Example 4.2 in Section 4).

- iv. Validate the solution.** We use Borel's law of large numbers to check the quality of solutions. For each n , let (y^*, u^*) denote the optimal solution (depending on n) found at step iii and draw a large random sample of size N_a for ξ . Let N_s be the number of times such that the event $G(y^*, u^*, \xi) \geq 0$ occurs. Simplifying, Borel's law of large numbers states that

$$\lim_{N_a \rightarrow \infty} \frac{N_s}{N_a} = P(G(y^*, u^*, \xi) \geq 0),$$

which allows to estimate *a posteriori*, for each n , the actual probability that the solution (y^*, u^*) (computed using KDE on n samples) verifies the constraint.

All tests have been performed on a laptop equipped with an Intel i7-4558U CPU running at 2.8 GHz and 8 GB of RAM. Throughout the three examples in Section 4, the performances are mostly constant. The number of iterations required by WORHP to converge does not depend on the number of samples n and it falls in the range of 5 to 10. On the other hand, the CPU time per iteration grows as n increases, varying from less than a tenth of a second when n is less than 100 to a maximum of 5 seconds for $n = 10\,000$.

*worhp.de

†projects.coin-or.org/Ipopt

4 | NUMERICAL TESTS

4.1 | Example 1: Chance constrained fuel load optimization of a simple three-stage launcher

Model. We first consider a simple model for a three-stage launcher. We analyze the vertical ascent of a rocket consisting in three sections, each one having its own fuel load and engine. During the flight, the vehicle will separate the stage as soon as the fuel load it contains is exhausted. The ODE systems describing the dynamics of the i th phase and the initial conditions are

$$\begin{cases} \dot{r}(t) = v(t), & t \in (t_{i-1}, t_i), & \text{(altitude)} \\ \dot{v}(t) = \frac{T_i}{m(t)} - g, & & \text{(speed)} \\ \dot{m}(t) = -\frac{T_i}{v_{ei}}, & & \text{(mass)} \\ (r(0), v(0), m(0)) = (0, 0, m_0). \end{cases} \quad (10)$$

For each phase i , we define the final time t_i , the engine thrust T_i , and the fuel speed v_{ei} , g being the gravitational acceleration of the Earth. The initial mass of the launcher is defined as the sum of the three stages fuel and structure, plus the payload

$$m_0 = \sum_{i=1}^3 (1 + k_i) m_{ei} + m_u,$$

where $\mathbf{k} = (k_1, k_2, k_3)$ and $\mathbf{m}_e := (m_{e1}, m_{e2}, m_{e3})$ are respectively the indexes and the fuel masses of the three stages. The initial mass m_0 has to satisfy the inequality $m_0 < \frac{T}{g}$ in order that the engine is powerful enough to overcome gravity. According to what has been said, the final time of each phase satisfies

$$t_1 = \frac{v_{e1} m_{e1}}{T_1}, \quad t_2 = t_1 + \frac{v_{e2} m_{e2}}{T_2}, \quad t_3 = t_2 + \frac{v_{e3} (m_{e3} + m_u)}{T_3}.$$

In the definition of t_3 , the payload m_u is summed to the fuel mass of the third stage. This allows the launcher to consume part of the payload in case the amount of fuel is not sufficient to satisfy the constraint on the final position. We set $\mathbf{T} := (T_1, T_2, T_3)$ and $(\mathbf{v}_e := v_{e1}, v_{e2}, v_{e3})$. In this simple model, for $0 \leq t \leq t_1$, the solution to the ODE system is

$$\begin{aligned} r(t) &= \left(v_{e1} t - \frac{v_{e1}^2 m(0)}{T_1} \right) \ln \left(\frac{m(0)}{m(t)} \right) + v_{e1} t - \frac{g}{2} t^2, \\ v(t) &= v_{e1} \ln \left(\frac{m(0)}{m(t)} \right) - gt, \\ m(t) &= \sum_{i=1}^3 (1 + k_i) m_{ei} + \bar{m}_u - \frac{T_1}{v_{e1}} t. \end{aligned}$$

For $t_1 < t \leq t_2$, one has

$$\begin{aligned} r(t) &= \left(v_{e1} t - \frac{v_{e1}^2 m(0)}{T_1} \right) \ln \left(\frac{m(0)}{m(t_1)} \right) + \\ &+ \left(v_{e2} (t - t_1) - \frac{v_{e2}^2 (m(t_1) - k_1 m_{e1})}{T_2} \right) \ln \left(\frac{m(t_1) - k_1 m_{e1}}{m(t)} \right) + \\ &+ v_{e1} t_1 + v_{e2} (t - t_1) - \frac{g}{2} t^2, \\ v(t) &= v_{e1} \ln \left(\frac{m(0)}{m(t_1)} \right) + v_{e2} \ln \left(\frac{m(t_1) - k_1 m_{e1}}{m(t)} \right) - gt, \\ m(t) &= \sum_{i=2}^3 (1 + k_i) m_{ei} + \bar{m}_u - \frac{T_2}{v_{e2}} (t - t_1), \end{aligned}$$

and if $t_2 < t \leq t_3$,

$$\begin{aligned}
 r(t) &= \left(v_{e1}t - \frac{v_{e1}^2 m(0)}{T_1} \right) \ln \left(\frac{m(0)}{m(t_1)} \right) + v_{e1}t_1 + \\
 &\quad + \left(v_{e2}(t - t_1) - \frac{v_{e2}^2(m(t_1) - k_1 m_{e1})}{T_2} \right) \ln \left(\frac{m(t_1) - k_1 m_{e1}}{m(t_2)} \right) + \\
 &\quad + \left(v_{e3}(t - t_2) - \frac{v_{e3}^2(m(t_2) - k_2 m_{e2})}{T_3} \right) \ln \left(\frac{m(t_2) - k_2 m_{e2}}{m(t)} \right) + \\
 &\quad + v_{e1}t_1 + v_{e2}(t_2 - t_1) + v_{e3}(t - t_2) - \frac{g}{2}t^2, \\
 v(t) &= v_{e1} \ln \left(\frac{m(0)}{m(t_1)} \right) + v_{e2} \ln \left(\frac{m(t_1) - k_1 m_{e1}}{m(t_2)} \right) + \\
 &\quad + v_{e3} \ln \left(\frac{m(t_2) - k_2 m_{e2}}{m(t)} \right) - gt, \\
 m(t) &= (1 + k_3)m_{e3} + \bar{m}_u - \frac{T_3}{v_{e3}}(t - t_2).
 \end{aligned}$$

Before defining the stochastic optimization problem associated to this model, let us define the reference deterministic optimization problem

$$\begin{cases} \min_{\mathbf{m}_e \in \mathbb{R}_+^3} \sum_{i=1}^3 (1 + k_i)m_{ei} + m_u, \\ M_u(\mathbf{m}_e) \geq m_u, \end{cases} \quad (11)$$

where the constraint function is defined as

$$M_u(\mathbf{m}_e) := m(t_3(\mathbf{m}_e)) - k_3 m_{e3}.$$

For a given $\mathbf{m}_e$, $t_3(\mathbf{m}_e)$ is the solution to the equation obtained by imposing a constraint on the apogee of the launcher target orbit $r(t_3) + \frac{v^2(t_3)}{2g} = \omega_f$. Table 1 sums up the choice of parameters. The optimal solution found is $m_{e1}^* \simeq 0.2153$, $m_{e2}^* \simeq 0.1838$, and $m_{e3}^* \simeq 0.0774$, with a corresponding optimal cost $\sum_{i=1}^3 (1 + k_i)m_{ei}^* + \bar{m}_u \simeq 1.0241$. Figure 1 shows the corresponding optimal trajectory.

Problem statement. Let us now suppose that the parameters $\mathbf{T}$, $\mathbf{k}$, and $\mathbf{v}_e$ are arrays of uniformly distributed random variables. For example, for each i in $\{1, 2, 3\}$, this implies $T_i \sim U(T_{i-}, T_{i+})$, where $T_{i-} := \bar{T}_i(1 - \Delta T_i)$, $T_{i+} := \bar{T}_i(1 + \Delta T_i)$, and $\bar{T}_i$ denotes the expected value. We also define $\bar{\mathbf{T}} := (\bar{T}_1, \bar{T}_2, \bar{T}_3)$ and $\Delta \mathbf{T} := (\Delta T_1, \Delta T_2, \Delta T_3)$. The same properties and definitions hold for $\mathbf{k}$ and $\mathbf{v}_e$. If we want to write in form (7) the stochastic counterpart of problem (11), we have to keep in mind that the cost to be minimized now depends on the random array $\mathbf{k}$ and it has to be defined as an expectation

$$E \left[\sum_{i=1}^3 (1 + k_i)m_{ei} + m_u \right] = \sum_{i=1}^3 (1 + E[k_i])m_{ei} + m_u.$$

Since each k_i is a uniformly distributed random variable on the interval $[K_{i-}, K_{i+}]$ with expected value $\bar{k}_i$, we can write the cost as

$$\sum_{i=1}^3 (1 + E[k_i])m_{ei} + m_u = \sum_{i=1}^3 (1 + \bar{k}_i)m_{ei} + m_u.$$

This leads us to the stochastic optimization problem

$$\begin{cases} \min_{\mathbf{m}_e \in \mathbb{R}_+^3} \sum_{i=1}^3 (1 + \bar{k}_i)m_{ei} + m_u, \\ P(M_u(\mathbf{T}, \mathbf{k}, \mathbf{v}_e, \mathbf{m}_e) \geq m_u) \geq p, \end{cases} \quad (12)$$

TABLE 1 Parameters for the deterministic optimization

Parameter	T_i	k_i	v_{ei}	g	m_u	ω_f
Value	150	0.1	5	9.8	0.5	0.5

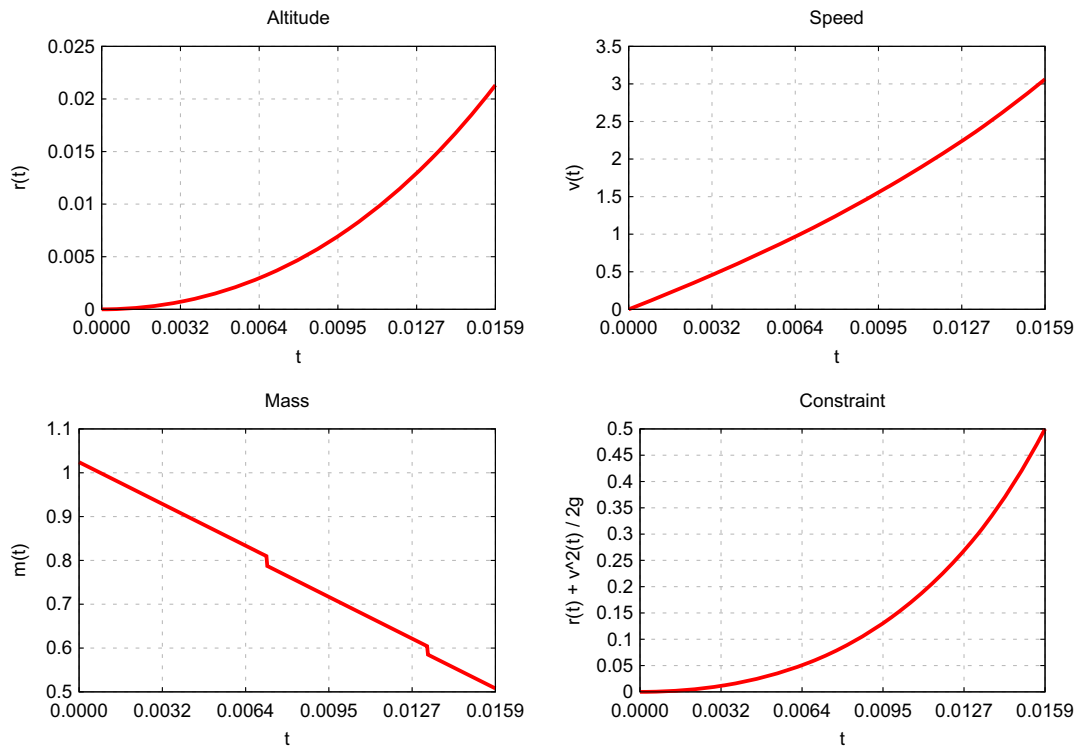


FIGURE 1 Plot of altitude, speed, mass, and constraint for the three-stage launcher [Colour figure can be viewed at wileyonlinelibrary.com]

TABLE 2 Additional parameters for the stochastic optimization

Parameter	p	$\bar{T}_i$	ΔT_i	$\bar{k}_i$	Δk_i	$\bar{v}_{ei}$	Δv_{ei}
Value	0.9	150	0.1	0.1	0.1	5	0.1

with a total of nine uniform random variables (three random arrays of dimension three), ie, $\mathbf{T}$, $\mathbf{k}$, and $\mathbf{v}_e$. The function $M_u(\mathbf{T}, \mathbf{k}, \mathbf{v}_e, \mathbf{m}_e)$ depends on the random arrays $\mathbf{T}$, $\mathbf{k}$, and $\mathbf{v}_e$, and on the parameter $\mathbf{m}_e$

$$M_u(\mathbf{T}, \mathbf{k}, \mathbf{v}_e, \mathbf{m}_e) := m(t_3(\mathbf{T}, \mathbf{k}, \mathbf{v}_e, \mathbf{m}_e)) - k_3 m_{e3}.$$

Table 2 summarizes the choice of parameters for this example.

Application of the method. Let us denote $F_{\mathbf{m}_e}(m_u)$ the CDF of the random variable M_u

$$F_{\mathbf{m}_e}(m_u) := \int_0^{m_u} f_{\mathbf{m}_e}(x) dx.$$

For each value of $\mathbf{m}_e$, we are able to produce an approximation $\hat{F}_{\mathbf{m}_e}$ of $F_{\mathbf{m}_e}$ via KDE (and quadrature) by drawing a sample from the random arrays $\mathbf{T}$, $\mathbf{k}$, and $\mathbf{v}_e$. Our problem becomes

$$\begin{cases} \min_{\mathbf{m}_e \in \mathbb{R}_+^3} \sum_{i=1}^3 (1 + \bar{k}_i) m_{ei} + m_u, \\ \hat{F}_{\mathbf{m}_e}(m_u) \leq 1 - p. \end{cases} \quad (13)$$

The procedure used for solving problem (13) is described in Section 3.2. As indicated for step ii in Section 3.2, we choose to use the SNR method (see (A4)) for computing the bandwidth combined with the Gaussian kernel.

Numerical results. Figure 2 shows the behavior of 10 sequences of optimal costs for $n \in \{100, 200, \dots, 10\,000\}$ and the corresponding rates of success computed *a posteriori* with $N_a = 10^5$. Figure 3 instead shows the average value and variance of the ten sequences previously shown for each n . For instance, for $n = 500$, the optimal solution is $m_{e1}^* \simeq 0.2222$, $m_{e2}^* \simeq 0.1835$, and $m_{e3}^* \simeq 0.1029$, with a corresponding optimal cost of $\sum_{i=1}^3 (1 + \bar{k}_i) m_{ei}^* + \bar{m}_u \simeq 1.0595$. This solution allows

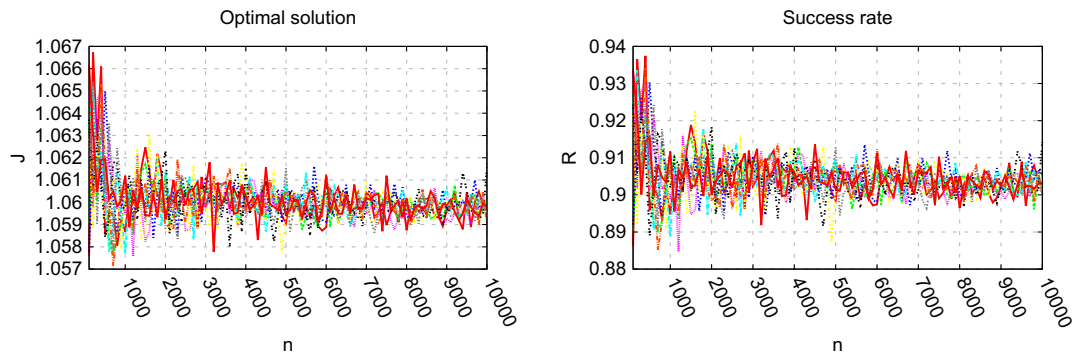


FIGURE 2 Plot of the optimal cost J and R as functions of n (10 simulations) [Colour figure can be viewed at wileyonlinelibrary.com]

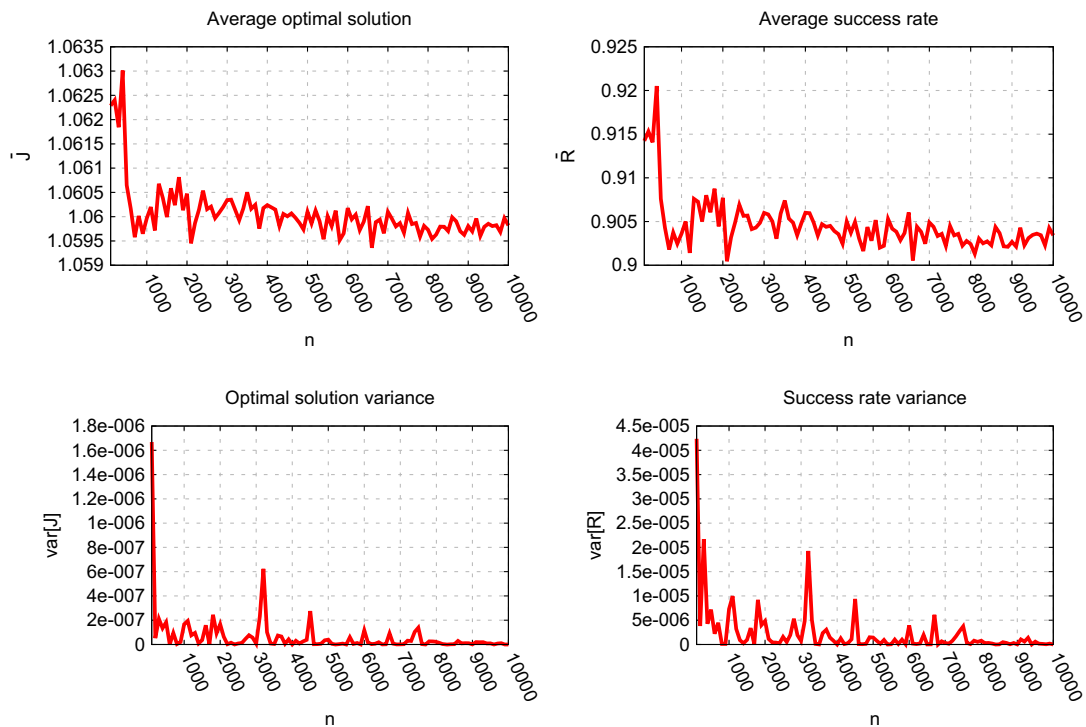


FIGURE 3 Plot of the average value and variance of optimal cost J and R as functions of n [Colour figure can be viewed at wileyonlinelibrary.com]

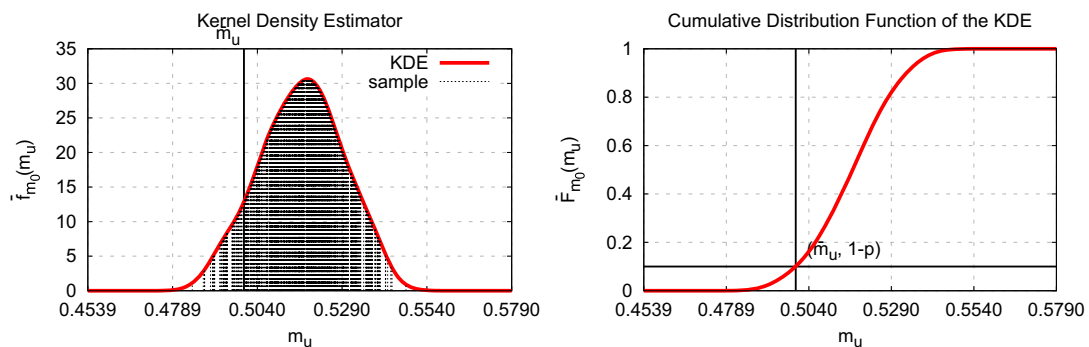


FIGURE 4 Plot of the kernel density estimator $\hat{f}$ of $M_u(T, \mathbf{m}_e^*)$ and its integral $\hat{F}$ [Colour figure can be viewed at wileyonlinelibrary.com]

us to deliver the payload $\bar{m}_u = 0.5$ with a success rate $R \simeq 90\%$ even if the maximum thrust T_i , the stage index k_i , and the fuel speed v_{ei} of each stage are subject to random uniform variations. Figure 4 shows the related plots. Table 3 compares

TABLE 3 Result comparison for extremal values of T , k , and v_e

Case	T_i	k_i	m_{ei}	m_0^*
Random	$\sim U(T_{i-}, T_{i+})$	$\sim U(K_{i-}, K_{i+})$	$\sim U(v_{ei-}, v_{ei+})$	1.0595
Best	T_{i+}	k_{i-}	v_{ei+}	0.9497
Worst	T_{i-}	k_{i+}	v_{ei-}	1.1246

TABLE 4 Result comparison for different values of n , p , and ΔT

n	p	ΔT_i Δk_i Δv_{ei}	h	m_{e1}	m_{e2}	m_{e3}	m_0^*	R
Stochastic								
10^4	0.8	0.5	0.0117	0.218	0.191	0.202	1.1714	0.7966
		0.25	0.00508	0.226	0.185	0.124	1.0887	0.8120
		0.1	0.00191	0.237	0.158	0.103	1.0477	0.8055
		0.01	0.00019	0.214	0.166	0.098	1.0259	0.8024
		0.001	0.00002	0.209	0.167	0.101	1.0239	0.8281
	0.9	0.5	0.01472	0.186	0.281	0.271	1.3115	0.9052
		0.25	0.00517	0.227	0.205	0.133	1.1215	0.8993
		0.1	0.00192	0.218	0.184	0.106	1.0596	0.9018
		0.01	0.00018	0.215	0.166	0.098	1.0271	0.9013
		0.001	0.00002	0.214	0.166	0.097	1.0241	0.9397
	0.995	0.5	no convergence					
		0.25	0.00579	0.261	0.214	0.197	1.2394	0.9960
		0.1	0.00202	0.220	0.212	0.106	1.0919	0.9951
		0.01	0.00019	0.216	0.168	0.098	1.0298	0.9959
		0.001	0.00002	0.212	0.169	0.096	1.0244	0.9996
	Deterministic							
	—			0.215	0.184	0.077	1.0241	—

the solution we just found for the stochastic optimization problem to the two solution we obtain from the deterministic one in the best and worst case. We observe again that the optimal mass of the stochastic problem is smaller than the one obtained in the worst deterministic case but bigger than the one of the best case. Table 4 shows the comparison between the solution of the deterministic problem 11 and its stochastic counterpart (12) when p is close to 1 and ΔT_i , Δk_i , and Δv_{ei} are close to 0. Unfortunately though, this method does not allow arbitrarily small values of ΔT_i , Δk_i , or Δv_{ei} . As reported in the table, when we do not provide enough variation to the sample, the success rate does not match the chosen probability. This is likely due to two issues related to the presence of the sample variance in (A4), and therefore to ΔT_i , Δk_i , or Δv_{ei} . First, if they are too small, the Gaussian distributions summed in (6) tend to superimpose over the same points and do not spread on the real axis. This adds probability mass outside the domain of the distribution to be estimated. A negligible manifestation of this phenomenon can be observed even with $\Delta T_i = \Delta v_{ei} = \Delta k_i = 0.1$ in Figure 4. Notice the space beneath the red graph on the left and right sides of the vertical sample lines. Secondly, because the bandwidth depends on the sample variance, the accuracy of the estimator might decrease if h is too small, as h appears as a denominator in (6). The results showed in Table 4 confirm that it is possible to increase the variance of the sample by increasing the number of random variables.

4.2 | Example 2: Chance constrained Goddard problem

We now apply the KDE technique to the Goddard problem. Formally, the structure of the model is the same as the example illustrated in the introduction. The vertical ascent of a launcher in one dimension is controlled by $u(t) \in [0, 1]$ (proportional to the thrust applied at time t). The main difference between the Goddard problem and (2) is the addition of the drag force to the dynamics. For the purpose of defining a probabilistic constraint, we consider the thrust T as the only random parameter and our objective is to maximize the final mass of the launcher while making sure that its altitude is higher than a given value ρ_f with a probability of at least p . In contrast with Example 1 that boiled down to a finite

dimensional optimization problem, a solution now consists in an optimal control function $u^* : \mathbf{R}_+ \rightarrow [0, 1]$ such that, if we apply u^* regardless of the value of T , the probability of the final altitude being greater than ρ_f is greater than p .

Model. The original formulation of the Goddard problem can be found in the work of Goddard.¹ We will consider a one-dimensional version of the one treated in the work of Bonnans et al.³⁸ The ODE system is

$$\begin{cases} \dot{r}(t) = v(t) & t \in [0, t_f] \\ \dot{v}(t) = \frac{T u(t) - A v^2(t) e^{-\kappa(r(t)-r_0)}}{m(t)} - \frac{1}{r^2(t)} & t \in [0, t_f] \\ \dot{m}(t) = -b u(t) & t \in [0, t_f] \\ (r(0), v(0), m(0)) = (r_0, 0, m_0), \end{cases}$$

where the final time $t_f > 0$ is free. The control function u belongs to $\mathcal{U}$, where

$$\mathcal{U} := \{u : \mathbf{R}_+ \rightarrow [0, 1] \subset \mathbf{R} \mid u \text{ is measurable}\}.$$

We will integrate the equations numerically by using the standard fourth-order Runge-Kutta method, as we did in the previous example, the control being now approximated by means of piecewise constant functions. Before defining our stochastic optimization problem, we first show the solution to the deterministic one

$$\begin{cases} \max_{(t_f, u) \in \mathbf{R}_+ \times \mathcal{U}} m(t_f), \\ r(t_f) \geq \rho_f. \end{cases} \quad (14)$$

Table 5 sums up the choice of parameters for this model. The optimal final time and cost found by WORHP are $t_f^* \simeq 0.1742$ and $m(t_f^*) \simeq 0.6297$. Figure 5 shows the corresponding optimal trajectory.

TABLE 5 Parameters for the deterministic optimization

Parameter	T	A	κ	b	r_0	m_0	ρ_f	n_t
Value	3.5	310	500	7	1	1	1.01	100

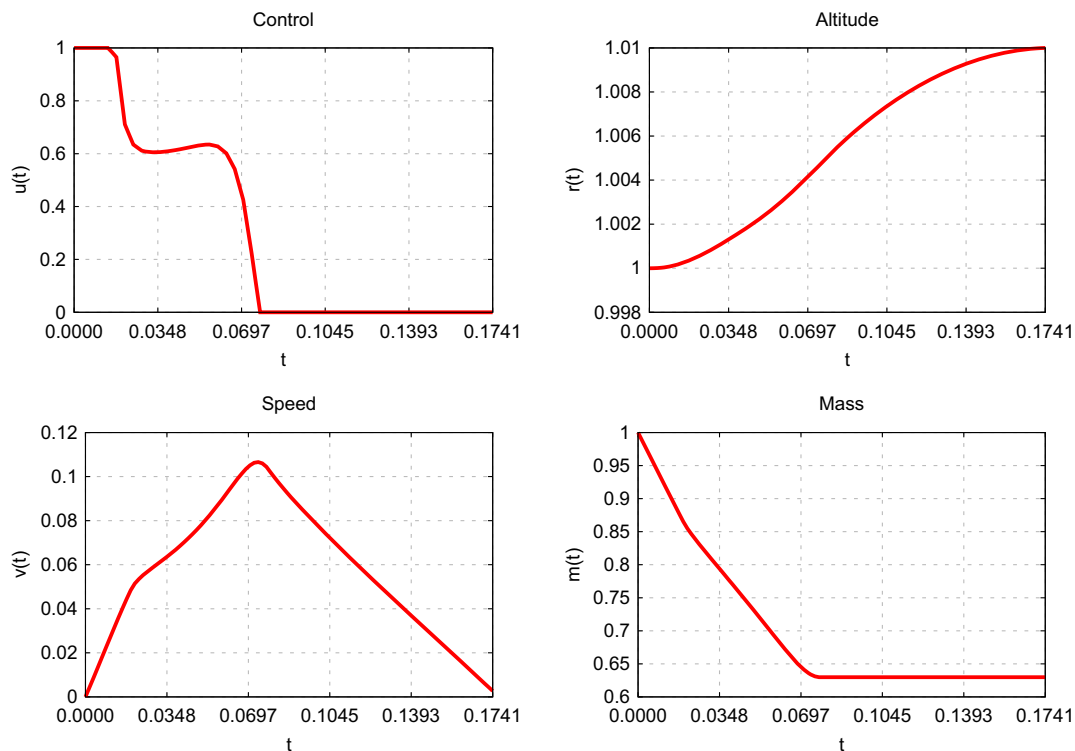


FIGURE 5 Plot of control, altitude, speed, and mass for the Goddard problem [Colour figure can be viewed at wileyonlinelibrary.com]

Problem statement. Our goal to reach at least the altitude ρ_f with a 90% probability while maximizing the final mass of the launcher. Keeping in mind that the cost to be minimized also depends on the random parameter T , it has to be defined as an expectation.

$$E[m(t_f)] = E\left[\int_0^{t_f} m_0 - \frac{T}{v_e} u(t) dt\right] = \int_0^{t_f} m_0 - \frac{E[T]}{v_e} u(t) dt.$$

We recall that T is a uniformly distributed random variable on the interval $[T_-, T_+]$ with expected value $\bar{T}$, so the cost is defined as

$$\bar{m}(t_f) := \int_0^{t_f} m_0 - \frac{\bar{T}}{v_e} u(t) dt.$$

This leads us to the stochastic optimization problem

$$\begin{cases} \max_{(t_f, u) \in \mathbb{R}_+ \times \mathcal{U}} \bar{m}(t_f), \\ P(r_f(t_f, u, T) \geq \rho_f) \geq p, \end{cases} \quad (15)$$

where $r_f(t_f, u, T)$ is the final altitude as a function of the random variable T , parameterized by u . Table 6 shows the choice of parameters defined in this section.

TABLE 6 Additional parameters for the stochastic optimization

Parameter	p	$\bar{T}$	ΔT
Value	0.9	3.5	0.1

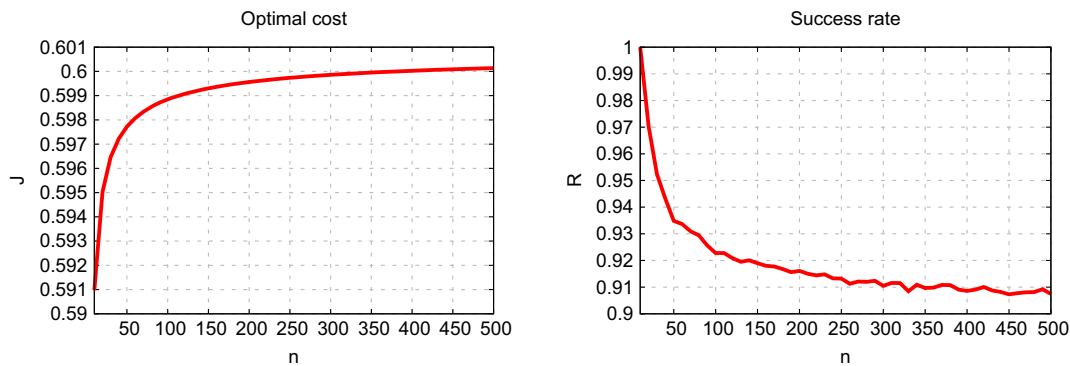


FIGURE 6 Plot of $m(t_f, u^*)$ and R as functions of n [Colour figure can be viewed at wileyonlinelibrary.com]

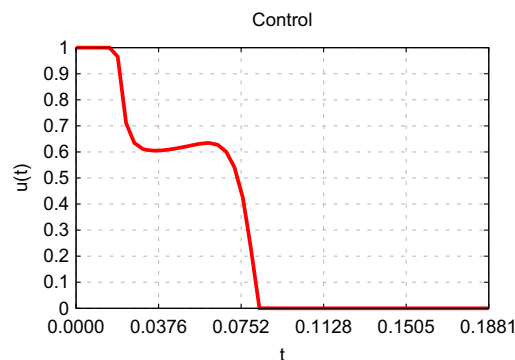


FIGURE 7 Optimal control for $n = 500$ [Colour figure can be viewed at wileyonlinelibrary.com]

Application of the method. By using the definition of the density function $f_{t_f,u}$ of the random variable $r_f(t_f, u, T)$ and using as previously an approximation by quadrature $\hat{F}_{t_f,u}$ of its CDF, we consider

$$\begin{cases} \max_{(t_f,u) \in \mathbb{R}_+ \times \mathcal{U}} \bar{m}(t_f), \\ \hat{F}_{(t_f,u)}(\rho_f) \leq 1 - p. \end{cases} \quad (16)$$

The procedure used for solving problem (16) is described in Section 3.2, with the only difference that, for this example, we do not take a random sample from variable T . Because we only have one random variable, we can take a uniform deterministic sample of T by dividing the interval $[T_-, T_+]$ into $n - 1$ subintervals. As before, we choose to use the SNR method (A4) to compute the bandwidth combined with the Gaussian kernel.

Numerical results. Figure 6 shows the behavior of the sequence of optimal costs for $n \in \{10, 20, 30, \dots, 500\}$ and the corresponding rate of success computed *a posteriori* with $N_a = 10^5$. For instance, for $n = 500$, the optimal final time is $t_f^* \simeq 0.1881$, with a corresponding cost $\bar{m}(t_f^*) \simeq 0.6001$ and a success rate $R = 90.81\%$. The corresponding optimal control u^* is shown in Figure 7, whereas estimations of the density of its CDF are shown in Figure 8. Table 7 and Figure 9 compare the solution we just found for the stochastic optimization problem to the two solutions we obtain from the deterministic one in the best and worst cases.

It can be seen how the solution to the chance constrained problem is slightly better than the one in the worst case, but still lower than the one corresponding to the best case. Interestingly, Figure 9 shows that the shape of the control strategy

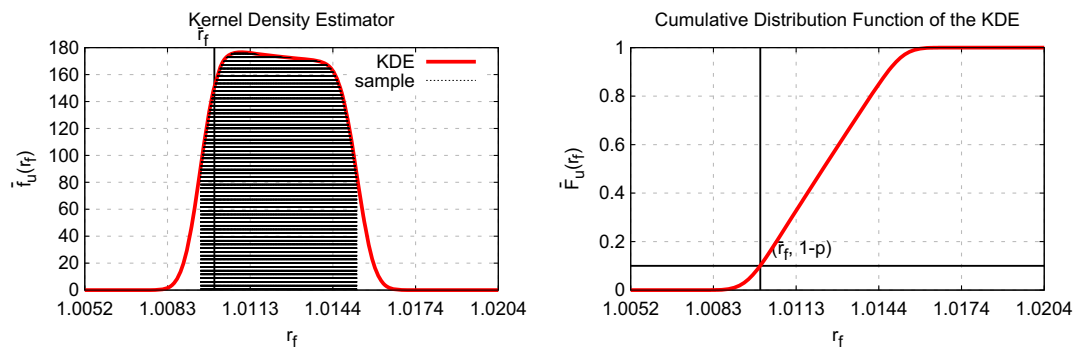


FIGURE 8 Plot of the kernel density estimator of $r_f(t_f^*, u^*, T)$ (left) and of its CDF (right) [Colour figure can be viewed at wileyonlinelibrary.com]

TABLE 7 Result comparison for extremal values of T

Case	T	t_f^*	$\bar{m}(t_f^*)$
Random	$\sim U(T_-, T_+)$	0.1881	0.6001
Best	T_+	0.1613	0.6584
Worst	T_-	0.1902	0.5928

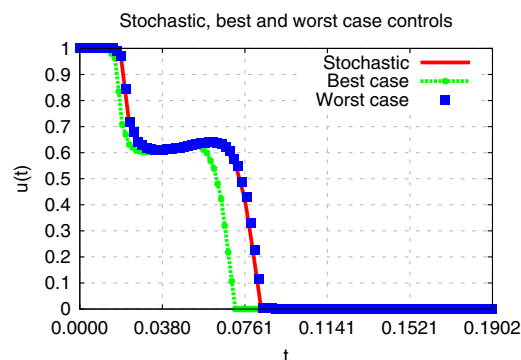


FIGURE 9 Comparison between stochastic, best, and worst case controls [Colour figure can be viewed at wileyonlinelibrary.com]

TABLE 8 Result comparison for different values of n , p , and ΔT

n	p	ΔT	h	$\bar{m}(t_f^*)$	R
Stochastic					
500	0.8	0.5	0.00517	0.4808	0.7980
		0.25	0.00155	0.5701	0.8011
		0.1	0.00048	0.6085	0.7996
		0.05	0.00022	0.6197	0.7999
		0.025	0.00010	0.6267	0.6879
	0.9	0.5	0.00813	0.3866	0.9096
		0.25	0.00185	0.5419	0.9090
		0.1	0.00051	0.6001	0.9090
		0.05	0.00023	0.6162	0.8929
		0.25	0.00011	0.6222	0.9479
		0.5	0.02127	0.1538	1.0000
		0.25	0.00271	0.4728	1.0000
		0.1	0.00057	0.5827	1.0000
		0.05	0.00024	0.6075	1.0000
		0.025	0.00011	0.6186	1.0000
Deterministic					
—				0.6297	—

does not change much between the three cases, and the main difference lies in the optimal value for the final time t_f^* . Table 8 shows the comparison between the solution of the deterministic problem (14) and its stochastic counterpart (15) when p is close to 1 and T_i is close to 0. For the results in the table, we set the initial guess for u equal to the optimal solution found for the deterministic problem (see Figure 5).

4.3 | Complex three-stage launcher with one decision variable and two random variables

We eventually address the more complex model of a real space launcher and consider the following percentile optimization problem:

$$\begin{cases} \min_{\mu \in \mathbb{R}} \mu, \\ P(G(\xi) \leq \mu) \geq p. \end{cases} \quad (17)$$

There are two random parameters. The specific impulse I_{sp_3} and the index K_3 of the third stage. As a function of both I_{sp_3} and K_3 , the optimal fuel mass of the third stage is also random, and our goal is to compute the 0.9 percentile of its distribution.

Model. The inertial equatorial frame coordinate system $\mathcal{F} := (O, \mathbf{i}, \mathbf{j}, \mathbf{k})$ is defined Figure 10B. O is the center of the Earth, $\mathbf{k}$ is the versor of Earth rotation axis directed toward north, $\mathbf{i}$ is the versor that belongs to Earth equatorial plane and points toward the Greenwich meridian, and $\mathbf{j} := \mathbf{k} \times \mathbf{i}$ completes the coordinate system. In this coordinate system, we define

$$\begin{aligned} \mathbf{x} &:= x\mathbf{i} + y\mathbf{j} + z\mathbf{k} \\ \mathbf{v} &:= \dot{\mathbf{x}} := v_x\mathbf{i} + v_y\mathbf{j} + v_z\mathbf{k} \\ \mathbf{v}_r(\mathbf{v}, \mathbf{x}) &:= \mathbf{v} - (0, 0, \Omega) \times \mathbf{x} \end{aligned}$$

to be respectively the position, the velocity, and the relative velocity of the launcher center of mass G , where Ω is the Earth's angular speed. We denote (ϕ, λ, h) the geographic coordinates of G , as shown in Figure 10A. ϕ is the latitude, λ the longitude, and h the height. The conversion formulas between cartesian and geographic coordinates can be found in the work of Gerdan and Deakin.³⁹ There is a number of variables and parameters attached to the launcher. We first define its longitudinal axis. This axis passes through G and points toward the edge of the launcher (see Figure 10C). We define the following angles.

- The launch azimuth.* ψ is the angle between the perpendicular line to the longitudinal axis at the initial position directed toward north and the orbit plane. The launch azimuth must satisfy the following equation to allow the

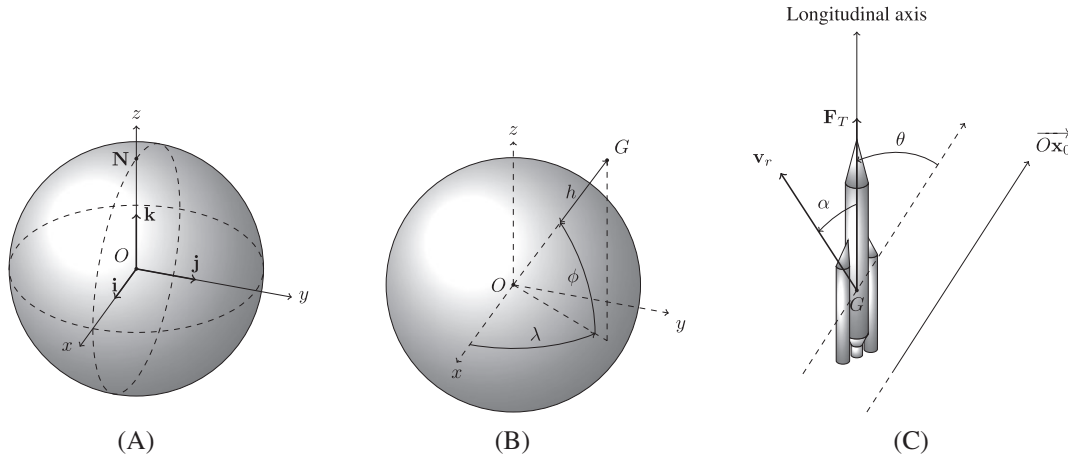


FIGURE 10 Reference's frame. A, Coordinates of G ; B, The coordinate system $\mathcal{F}$; C The angles θ and α

launcher to reach the target orbit inclination. $\psi = \arcsin(\cos(i)/\cos(\phi_0))$, meaning that the inclination i must be greater than the launch site latitude ϕ_0 .

- ii. *The angle of attack.* α is the angle between the longitudinal axis and the relative velocity $\mathbf{v}_r$ measured in the orbit plane.
- iii. *The pitch angle.* θ is the angle between the longitudinal axis and the vector $\overrightarrow{OX_0}$ measured in the orbit plane. The orbit plane is the plane of the ellipse that defines the geostationary transfer orbit, it is characterized by two angles, ie, the longitude of the ascending node and the angle of inclination with respect to the equatorial plane of the Earth. Not all the inclinations can be reached from a given launch site. The location has to be a point inside the target orbit plane.

Moreover, we call β_i , I_{sp_i} , and S_i respectively the mass flow rate, the specific impulse, and the area of the nozzle's section of the i th stage engine. Furthermore, we denote A_i the as area of the i th stage reference surface involved in the computation of the drag force. We denote m as the total mass of the vehicle. Depending on the flight phase, it is the sum of some of the payload m_p , payload case m_c , the fairing m_f , the i th stage fuel $m_{ei}(t)$ at time t (the initial fuel mass of each stage is $m_{ei0} := m_{ei}(t_0)$, $i \in \{1, 2, 3\}$), and the i th stage structure m_{si} equal to $m_{si} := K_i m_{ei0}$ (K_i is the i th stage index). The launcher is subject to three forces, ie, the force $\mathbf{F}_G$ due to gravity, the drag force $\mathbf{F}_D$, and the thrust force $\mathbf{F}_T$. See Appendix B for a detailed description. In cartesian coordinates, the equations of motion write

$$\begin{cases} \dot{\mathbf{x}}(t) = \mathbf{v}(t), \\ m(t)\dot{\mathbf{v}}(t) = \mathbf{F}_G(m(t), \mathbf{x}(t)) + \mathbf{F}_D(\mathbf{x}(t), \mathbf{v}(t)) + \mathbf{F}_T(\theta(t), \mathbf{x}(t), \mathbf{v}(t)), \\ \dot{m}(t) = -\beta. \end{cases} \quad (18)$$

The direction of the launcher is controlled by acting on the pitch angle θ at any time t . For a given position $\mathbf{x}$ and velocity $\mathbf{v}$, the perigee and apogee of the associated orbit are given by

$$L_p(\mathbf{x}, \mathbf{v}) = (1 - \varepsilon(\mathbf{x}, \mathbf{v}))a(\mathbf{x}, \mathbf{v}) - R_e \quad L_a(\mathbf{x}, \mathbf{v}) = (1 + \varepsilon(\mathbf{x}, \mathbf{v}))a(\mathbf{x}, \mathbf{v}) - R_e,$$

where ε is the eccentricity of the orbit

$$\varepsilon(\mathbf{x}, \mathbf{v}) = \sqrt{1 - \frac{||\mathbf{x} \times \mathbf{v}||^2}{\mu_0 a(\mathbf{x}, \mathbf{v})}},$$

and a is the semimajor axis

$$a(\mathbf{x}, \mathbf{v}) = \frac{1}{\frac{2}{||\mathbf{x}||} - \frac{||\mathbf{v}||^2}{\mu_0}}.$$

The flight sequence consists in several phases, we will use the following notation to denote the duration and the final time of each flight phase. t_0 is the initial time, τ_i is the duration of the phase i , τ_{ij} is the duration of the sub-phase i,j , t_i is the final time of the phase i , and t_{ij} is the final time of the subphase i,j .

Phase 1. The launch azimuth is fixed at value ψ and the initial position at the geographic coordinates (ϕ_0, λ_0, h_0) . During this phase, the mass of the launcher is $m(t) = m_p + m_c + m_f + \sum_{i=1}^3 (1 + K_i) m_{ei}(t)$, $t \in [t_0, t_1)$. The engine of the first stage is ignited and the launcher accelerates vertically (ie, with the same direction of $\vec{OG}$), leaving the service structure. The pitch angle for this subphase is $\theta(t) \equiv 0$, $t \in [t_0, t_{1.1})$. Then, the launcher rotates with constant speed changing its orientation

$$\theta(t) = \frac{\theta_1}{\tau_{1.2}}(t - t_{1.1}), \quad t \in [t_{1.1}, t_{1.2}).$$

After the tilt, the direction of thrust is set to the final value of the previous subphase until the angle of incidence α vanishes

$$\theta(t) = \theta_1, \quad t \in [t_{1.2}, t_{1.3}),$$

where $t_{1.3} := \min_{t \in (t_{1.2}, \infty)} \{t \mid \alpha(t) = 0\}$.

The final subphase is a zero incidence flight until exhaustion of the first stage fuel, ie, $\tau_1 = m_{e10}/\beta_1$. This subphase ends with the separation of the first stage.

Phase 2. At the beginning of this phase, the mass of the launcher is $m(t) = m_p + m_c + m_f + \sum_{i=2}^3 (1 + K_i) m_{ei}(t)$, $t \in [t_1, t_{2.1})$. The second stage engine ignites. This subphase ends with the release of the fairing as soon as the heat flux decreases to a given value

$$\theta(t) = \theta_2 + \theta'_2(t - t_1), \quad t \in [t_1, t_{2.1}),$$

where $t_{2.1} := \min_{t \in (t_1, \infty)} \{t \mid \Gamma(\mathbf{x}(t), \mathbf{v}(t)) \leq \Gamma^*\}$

and where $\Gamma(\mathbf{x}, \mathbf{v}) = \frac{1}{2} \rho(\mathbf{x}) \|\mathbf{v}_r(\mathbf{x}, \mathbf{v})\|^3$ represents the heat flux. The mass changes to $m(t) = m_p + m_c + \sum_{i=2}^3 (1 + K_i) m_{ei}(t)$, $t \in [t_{2.1}, t_2)$. The flight continues without fairing until complete consumption of the fuel in the second stage, ie, $\tau_2 = m_{e20}/\beta_2$. This subphase ends with the jettison of the second stage, and the pitch angle is $\theta(t) = \theta_2 + \theta'_2 \tau_{2.1} + \theta'_2(t - t_1)$, $t \in [t_1, t_{2.1})$.

Phase 3. During this phase, the mass of the launcher is $m(t) = m_p + m_c + (1 + K_3) m_{e3}(t)$, $t \in [t_{2.2}, t_f)$. The third stage engine ignites, and this phase ends when fuel is exhausted, ie, $\tau_3 = m_{e30}/\beta_3$. At the final time $t_f := t_3$, both position and velocity must belong to the target orbit

$$\begin{aligned} \theta(t) &= \theta_3 + \theta'_3(t - t_2), \quad t \in [t_{2.2}, t_f) \\ L_p(\mathbf{x}(t_f), \mathbf{v}(t_f)) &= L_p^* \\ L_a(\mathbf{x}(t_f), \mathbf{v}(t_f)) &= L_a^*. \end{aligned}$$

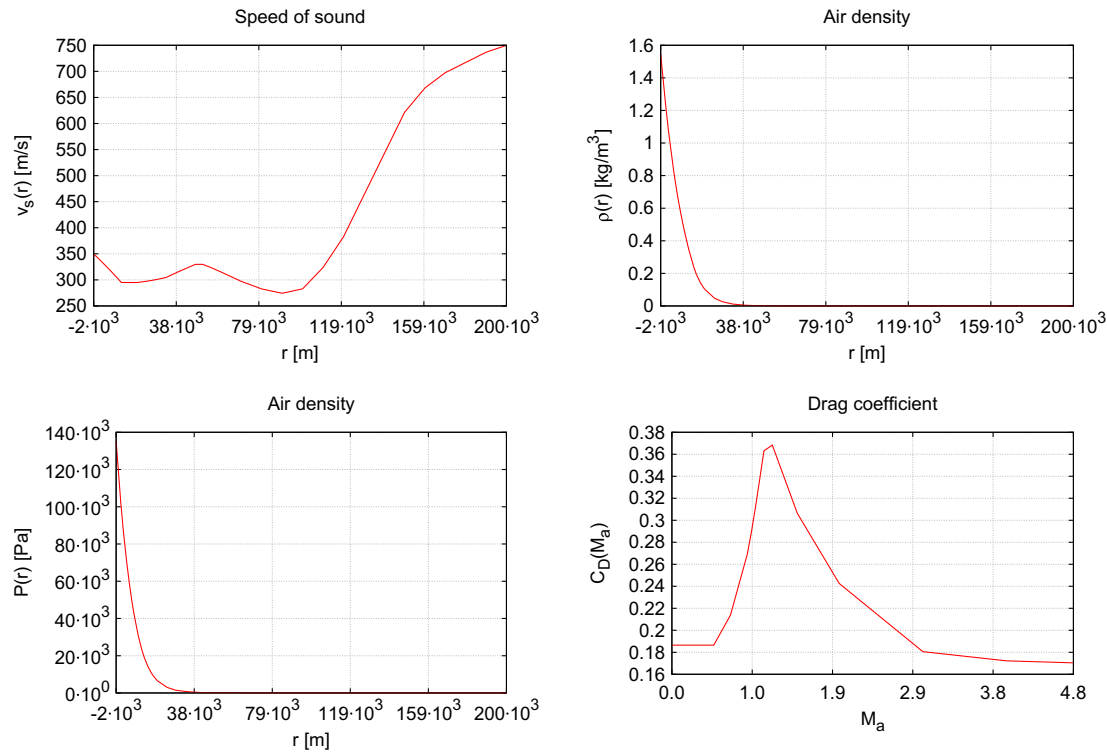
We can now formulate the following deterministic optimization problem:

$$\begin{cases} \min_{(m_{e30}, \theta_1, \theta_2, \theta'_2, \theta_3, \theta'_3) \in \mathbb{R}_+^6} m_{e30}, \\ L_p(m_{e30}, \theta_1, \theta_2, \theta'_2, \theta_3, \theta'_3) = L_p^*, \\ L_a(m_{e30}, \theta_1, \theta_2, \theta'_2, \theta_3, \theta'_3) = L_a^*, \end{cases} \quad (19)$$

where, with a slight abuse of notation, the functions L_p and L_a denote, respectively, the perigee and apogee associated to the final state $(\mathbf{x}(t_f), \mathbf{v}(t_f))$. Table 9 summarizes the choice of all the fixed parameters of the problem, whereas Figure 11 shows the profile of the speed of sound, the air density, the atmospheric pressure (each one depending on altitude), and the drag coefficient (depending on the Mach number). With this choice of the duration of the first two flight phases, the fuel load of the corresponding stages can easily be computed (see Table 10) because of the relation $m_{ei} = \beta_i \tau_i$ for $i \in \{1, 2, 3\}$. The parameters for the Earth and the flight sequence are defined in Tables 11 and 12, respectively. The optimal values found by WORHP for the optimization variables are reported in Table 13 and Figure 12 shows the corresponding optimal trajectory. The ODE system (18) is integrated by using the Fortran 90 subroutine DOP853 described in the work of Hairer et al.⁴⁰

TABLE 9 Mechanical and structural parameters

Fairing		Case		Payload	
m_f	1.100 kg	m_c	858.86 kg	m_p	4500 kg
Stage 1		Stage 2		Stage 3	
K_1	0.13	K_2	0.13	K_3	0.13
β_1	1896.58 kg/s	β_2	273.49 kg/s	β_3	42.18 kg/s
I_{sp1}	345.32 s	I_{sp2}	349.4 s	I_{sp3}	450.72 s
S_1	7.18 m ²	S_2	5.16 m ²	S_3	1.97 m ²
A_2	17.35 m ²	A_3	17.35 m ²	A_1	17.35 m ²

**FIGURE 11** Speed of sound v_s , air density ρ , atmospheric pressure P , and drag coefficient C_D [Colour figure can be viewed at wileyonlinelibrary.com]**TABLE 10** Values for the initial fuel masses

m_{e10}	278797.26 kg
m_{e20}	60714.78 kg

TABLE 11 Earth's parameters

Ω	$7.292155 \cdot 10^{-5}$ rad/s
R_p	6356752 m
R_e	6378137 m
μ_0	$3.986005 \cdot 10^{14}$ m ³ /s ²
J_2	$1.08263 \cdot 10^{-3}$
g_0	9.80665 m/s ²

Problem statement. Let $m_{e30}(m_p, \xi)$ be the value function of (19), depending on $\xi := (I_{sp3}, K_3)$ and the dimensioning parameter m_p . Consider the following chance constrained optimization problem:

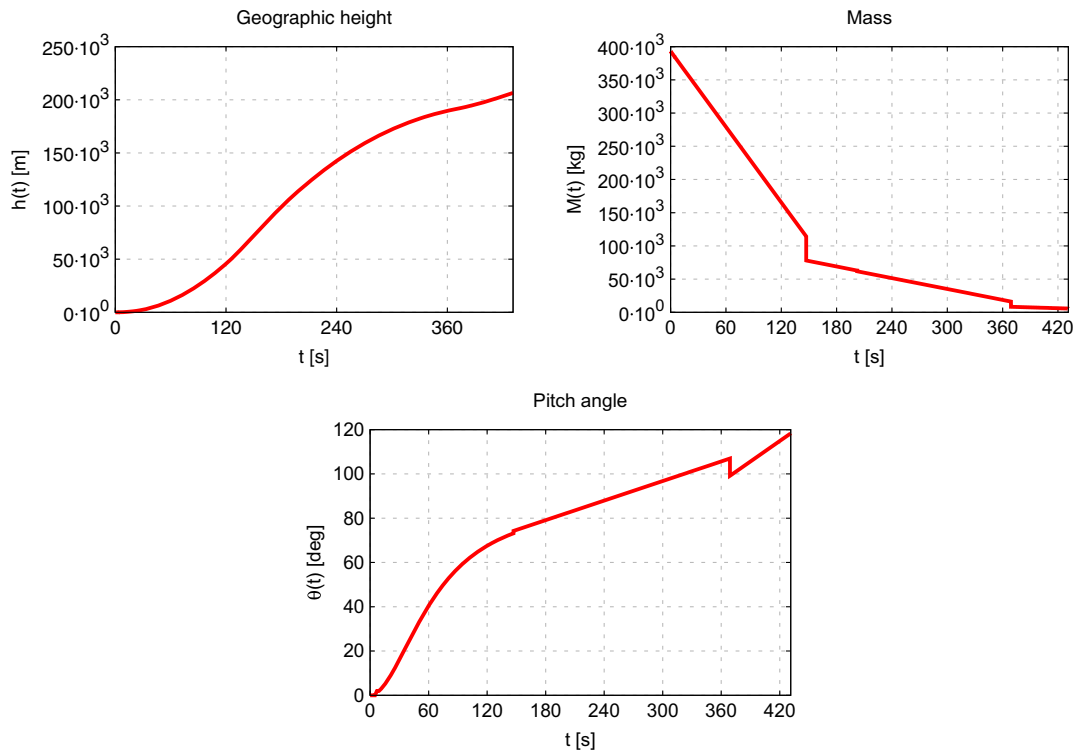
$$\begin{cases} \min_{\mu \in \mathbb{R}_+} \mu, \\ P(M_{e30}(m_p, \xi) \geq \mu) \geq p, \end{cases} \quad (20)$$

TABLE 12 Parameters for the flight sequence

Phase 1	Subphase 1.1	t_0	0 s
		ψ	90 deg
		ϕ_0	5.159722 deg
	Subphase 1.2	λ_0	-52.650278 deg
		h_0	0 m
		$\tau_{1.1}$	5 s
Phase 2	Subphase 2.1	$\tau_{1.2}$	2 s
		τ_1	147 s
		Γ^*	1135 W/m ²
Phase 3		τ_2	222 s
		L_p^*	200000 m
		L_a^*	35786000 m

TABLE 13 Optimal values for the free variables

m_{e30}	2627.1511 kg
θ_1	1.98164037 deg
θ_2	74.24468871 deg
θ_2'	0.14736836 deg/s
θ_3	99.15421943 deg
θ_3'	0.30801744 deg/s

**FIGURE 12** Result of the three-stage launcher optimization [Colour figure can be viewed at wileyonlinelibrary.com]

where I_{sp_3} and K_3 are uniformly distributed random variables, respectively, on the intervals $[I_{sp_3-}, I_{sp_3+}]$ and $[k_{3-}, k_{3+}]$, with expected values $\bar{I}_{sp_3}$ and $\bar{K}_3$. Here, $I_{sp_3-} := \bar{I}_{sp_3}(1 - \Delta I_{sp_3})$, $I_{sp_3+} := \bar{I}_{sp_3}(1 + \Delta I_{sp_3})$, and similar definitions hold for K_3 . Note that (20) matches the definition of the percentile optimization problem (17). The problem depends on two dimensioning parameters, ie, the payload m_p and the probability of success p . Table 14 shows the choice of parameters defined in this section. A crucial difference between this problem and the ones treated previously is that the decision variable is separated

TABLE 14 Additional parameters for the stochastic optimization

Parameter	p	$\bar{I}_{sp3}$	ΔI_{sp3}	$\bar{K}_3$	ΔK_3
Value	0.9	450.72 [s]	0.1	0.13	0.1

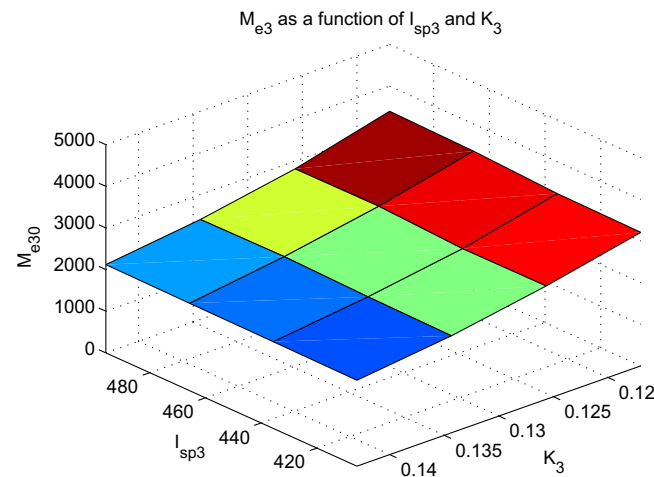
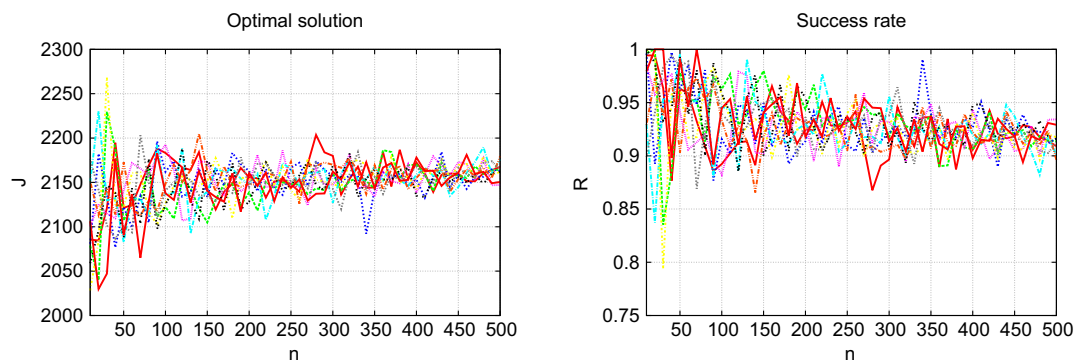
from the random ones. More precisely, y and ξ being the decision and the random variables, we can rewrite the chance constraint according to

$$P(D(\xi) \leq E(\mu)) \geq p,$$

which allows us to improve drastically the solver performances by precomputing (for a given value of m_p) the function $m_{e30}(m_p, \xi)$ at given grid values of the random variables ξ . Figure 13 shows the plot of $m_{e30}(m_p, \xi)$ as a function of ξ for our choice of m_p (see Table 9). The function has been evaluated at 16 values of ξ on an equally partitioned grid on the set $[I_{sp3-}, I_{sp3+}] \times [k_{3-}, k_{3+}]$. The values in between gridpoints are obtained via bilinear interpolation. We also recall that, because the constraint function is parameterized by the payload m_p , every change in its value would require a new computation of m_{e30} at grid values. For all the values of ξ in $[I_{sp3-}, I_{sp3+}] \times [k_{3-}, k_{3+}]$, the solver WORHP was able to compute an optimal control, allowing the launcher to reach its final orbit while minimizing the initial mass.

Application of the method. In order to use the KDE, we have to reformulate the chance constraint using the approximated CDF $\hat{F}_{m_p}$ of the random variable $m_{e30}(m_p, \xi)$

$$\begin{cases} \min_{\mu \in \mathbb{R}_+} \mu, \\ \hat{F}_{m_p}(\mu) \geq 1 - p. \end{cases} \quad (21)$$

**FIGURE 13** Plot of the third stage optimal fuel mass as a function of ξ [Colour figure can be viewed at wileyonlinelibrary.com]**FIGURE 14** Plot of μ^* and R as functions of n (10 simulations) [Colour figure can be viewed at wileyonlinelibrary.com]

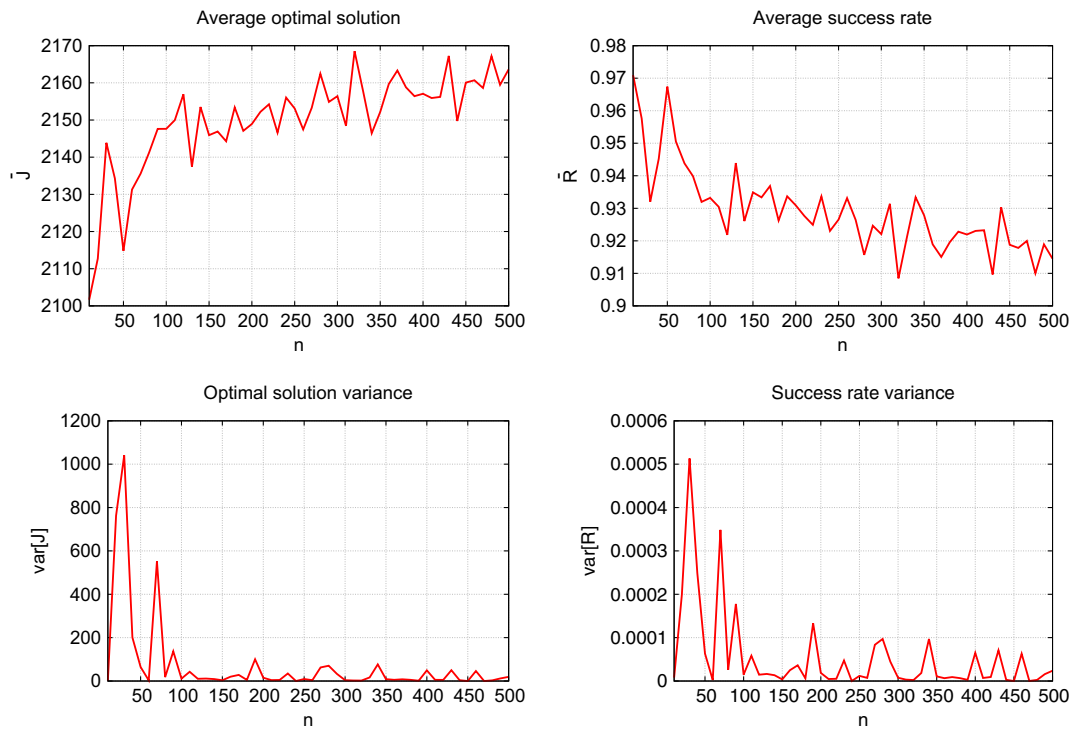


FIGURE 15 Plot of the average value and variance of μ^* and R as functions of n [Colour figure can be viewed at [wileyonlinelibrary.com](#)]

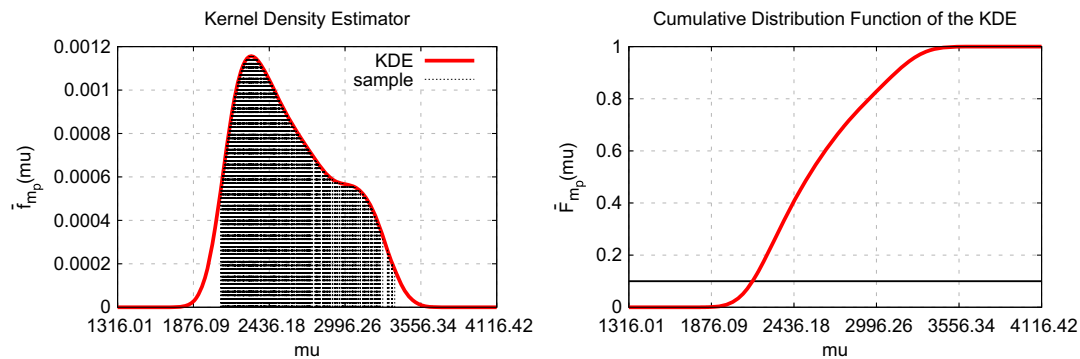


FIGURE 16 Plot of the kernel density estimator $\hat{f}_{m_p}$ of $M_{e_{30}}$ and its integral $\hat{F}$ [Colour figure can be viewed at [wileyonlinelibrary.com](#)]

As explained earlier, the remarkable feature of the problem is that, in contrast with the previous examples, the PDF estimator does not depend on the optimization parameter μ . We again use a Gaussian kernel together with the SNR bandwidth.

Numerical results. Figures 14 to 15 show the behavior of 10 sequences of optimal costs, for $n \in \{10, 20, 30, \dots, 500\}$, and the corresponding rate of success computed *a posteriori* with $N_a = 10^5$. For example, for $n = 500$, the optimal cost is $\mu^* \simeq 2162.78$ and the success rate is $R \simeq 91.83\%$. Figure 16 shows the related plots.

5 | CONCLUSION

The performances of the proposed KDE approach (coupled with a nonlinear problem solver) depends on a variety of factors, including the structure of the problem. As illustrated on the last example treated in Section 4, whether the decision variables and controls are separable from the random variables or not has a strong impact on the method, both from the theoretical and computational point of view. The bandwidth selection strategy also plays an important role. Some of the most refined methods to compute the bandwidth might require the minimization of an error function. The quadrature

formula used for the numerical integration of the density estimator, the discretization as an optimization problem, and the choice of the optimization solver itself strongly influence the results.

Throughout this paper, we showed how chance constrained optimization can be relevant to solve robust optimization and optimal control problems, especially when the traditional deterministic techniques like the worst-case analysis cannot be applied because they are not designed to take into account unfeasible solutions. In spite of a not yet complete theoretical framework, the numerical results provided by KDE are very promising. Even better results might be obtained by improving the computation of bandwidth h , for example, by substituting the second derivative f'' of the unknown density in (A2) with some tailored approximation. This so-called *plug-in* method is explained in detail in the work of Sheather.³³ Such a method can increase the accuracy of the estimator $\hat{f}$, but as it involves more complex operations for the computation of h compared to the SNR bandwidth (A4), we decided to implement the latter in our tests to preserve good performances. Pairing KDE with a robust NLP solver-like WORHP has proven solid enough to handle the three chance constrained optimization problems treated in Section 4. It is our hope that this paper will foster future research along this line. Further work includes comparing the proposed KDE method with other approaches such as those cited in Section 2 (scenario approach, back-mapping, stochastic Arrow-Hurwicz, etc) when all are applicable.

ACKNOWLEDGEMENTS

The authors want to thank Pierre Carpentier from ENSTA-Paristech for many fruitful discussions on stochastic optimization.

This work is partially supported by the EU under the 7th Framework Programme Marie Curie Initial Training Network “FP7-PEOPLE-2010-ITN”, SADCO project, GA number 264735-SADCO. For the third and fifth author, also supported by the iCODE Institute project funded by the IDEX Paris-Saclay, ANR-11-IDEX-0003-02.

ORCID

J.-B. Caillaud  <http://orcid.org/0000-0002-1719-2016>

REFERENCES

1. Goddard RH. A method of reaching extreme altitudes. *Smithson Misc Collect*. 1921;71(2):2-69.
2. Agrachev AA, Sachkov Y. *Control Theory from the Geometric Viewpoint*. Berlin, Germany: Springer-Verlag Berlin Heidelberg; 2004.
3. Vinter RB. *Optimal Control*. Berlin, Germany: Springer; 2000.
4. Wald A. Contributions to the theory of statistical estimation and testing hypotheses. *Ann Math Stat*. 1939;10(4):299-326.
5. Berstimas D, Sim M. The price of robustness. *Oper Res*. 2004;52(1):35-53.
6. Serra R. *Opérations de Proximité en Orbite: Évaluation du Risque de Collision et Calcul de Manœuvres Optimales Pour Lévitement et le Rendez-vous* [PhD thesis]. Toulouse, France: INSA Toulouse; 2015.
7. Serra R, Arzelier D, Joldes M, Rondepierre A. Probabilistic collision avoidance for long-term space encounters via risk selection. In: Bordeneuve-Guibé J, Drouin A, Roos C, eds. *Advances in Aerospace Guidance, Navigation and Control*. Cham Switzerland: Springer; 2015.
8. Sahin KH, Diwekar UM. Better optimization of nonlinear uncertain systems (BONUS): a new algorithm for stochastic programming using reweighting through kernel density estimation. *Ann Oper Res*. 2004;132(1-4):47-68.
9. Calfa BA, Grossmann IE, Agarwal A, Bury SJ, Wassick JM. Data-driven individual and joint chance-constrained optimization via kernel smoothing. *Comput Chem Eng*. 2015;78:51-69.
10. Nemirovski A, Shapiro A. Convex approximations of chance constrained programs. *SIAM J Optim*. 2006;17(4):969-996.
11. Zhang Y, Feng Y, Rong G. Data-driven chance constrained and robust optimization under matrix uncertainty. *Ind Eng Chem Res*. 2016;55(21):6145-6160.
12. Charnes A, Cooper WW, Symonds GH. Cost horizons and certainty equivalents: an approach to stochastic programming of heating oil. *Manag Sci*. 1958;4(3):235-263.
13. Prékopa A. On probabilistic constrained programming. In: *Proceedings of the Princeton Symposium on Mathematical Programming*; 1970; Princeton, NJ.
14. Prékopa A. Contributions to the theory of stochastic programming. *Math Program*. 1973;4(1):202-221.
15. Prékopa A. *Stochastic Programming*. Dordrecht, the Netherlands: Kluwer Academic Publishers; 1995.
16. Prékopa A. Probabilistic programming. In: Ruszczyński AAS, ed. *Stochastic Programming*. Vol. 10. New York, NY: Elsevier; 2003.
17. Dentcheva D. Optimization models with probabilistic constraints. In: Calafiore G, Dabbene F, ed. *Probabilistic and Randomized Methods for Design under Uncertainty*. London, UK: Springer; 2003.
18. Raik E. Qualitative research into the stochastic nonlinear programming problems. *Eesti NSV Teaduste Akademia Toimetised*. 1971;20:8-14.

19. van Ackooij W, Henrion R. Gradient formulae for nonlinear probabilistic constraints with Gaussian and Gaussian-like distributions. *SIAM J Optim*. 2014;24(4):1864-1889.
20. Uryasev S. Derivatives of probability functions and some applications. *Ann Oper Res*. 1995;56(1):287-311.
21. Marti K. Differentiation formulas for probability functions: the transformation method. *Math Program*. 1996;75(2):201-220.
22. Henrion R, Römisch W. Hölder and Lipschitz stability of solution sets in programs with probabilistic constraints. *Math Program*. 2004;100(3):589-611.
23. Calafiore GC, Campi MC. The scenario approach to robust control design. *IEEE Trans Autom Control*. 2006;51(5):742-753.
24. Campi MC, Garatti S. A sampling-and-discarding approach to chance-constrained optimization: feasibility and optimality. *J Optim Theory Appl*. 2015;148(2):257-280.
25. Carè A, Garatti S, Campi MC. Scenario min-max optimization and the risk of empirical costs. *SIAM J Optim*. 2015;25(4):2061-2080.
26. Wendt M, Li P, Wozny G. Nonlinear chance-constrained process optimization under uncertainty. *Ind Eng Chem Res*. 2002;41(15):3621-3629.
27. Klöppel M, Geletu A, Hoffmann A, Li P. Using sparse-grid methods to improve computation efficiency in solving dynamic nonlinear chance-constrained optimization problems. *Ind Eng Chem Res*. 2011;50(9):5693-5704.
28. Andrieu L, Cohen G, Vázquez-Abad FJ. Stochastic programming with probability constraints. 2007. arXiv:0708.0281v1.
29. Rosenblatt M. Remarks on some nonparametric estimates of a density function. *Ann Math Stat*. 1956;27(3):832-837.
30. Parzen E. On estimation of a probability density function and mode. *Ann Math Stat*. 1962;33(3):1065-1076.
31. Silverman BW. *Density Estimation for Statistics and Data Analysis*. New York, NY: Chapman & Hall; 1986.
32. Terrell GR, Scott DW. Variable kernel density estimation. *Ann Stat*. 1991;20(3):1236-1265.
33. Sheather SJ. Density estimation. *Stat Sci*. 2004;19(4):588-597.
34. Nadaraya ÉA. On non-parametric estimates of density functions and regression curves. *Theory Probab Its Appl*. 1965;10(1):186-190.
35. Silverman BW. Weak and strong uniform consistency of the kernel estimate of a density and its derivatives. *Ann Stat*. 1978;6(1):177-184.
36. Devroye L, Györfi L. *Nonparametric Density Estimation: The L_1 View*. New York, NY: Wiley; 1985.
37. Young DM, Gregory RT. *A Survey of Numerical Mathematics*. Boston, MA: Addison-Wesley; 1972.
38. Bonnans JF, Martinon P, Trélat E. Singular arcs in the generalized Goddard's problem. *J Optim Theory Appl*. 2008;139(2):439-461.
39. Gerdan GP, Deakin RE. Transforming cartesian coordinates X, Y, Z to geographical coordinates ϕ, λ, h . *Aust Surv*. 1999;44(1):55-63.
40. Hairer E, Nørsett SP, Wanner G. *Solving Ordinary Differential Equations I: Nonstiff Problems*. Berlin, Germany: Springer-Verlag Berlin Heidelberg; 1993. *Springer Series in Computational Mathematics*. Vol. 1-2.
41. Bhattacharya PK. Estimation of a probability density function and its derivatives. *Sankhyā: Indian J Stat*. 1967;29(4):373-382.
42. Giné E, Guillou A. Rates of strong uniform consistency for multivariate kernel density estimators. *Ann l'Institut Henri Poincare: Probab Stat*. 2002;38(6):907-921.
43. Prakasa Rao BLS. *Non-Parametric Functional Estimation*. Orlando, FL: Academic Press; 1983.
44. Ahmad I, Ameziane M. A general and fast convergent bandwidth selection method of kernel estimator. *J Nonparametric Stat*. 2007;19(4-5):165-187.
45. Minnotte MC. Achieving higher-order convergence rates for density estimation with binned data. *J Am Stat Assoc*. 1998;93(442):663-672.
46. Hodges JL, Lehmann EL. The efficiency of some nonparametric competitors of the t -test. *Ann Math Stat*. 1956;27(2):324-335.

How to cite this article: Caillau J-B, Cerf M, Sassi A, Trélat E, Zidani H. Solving chance constrained optimal control problems in aerospace via kernel density estimation. *Optim Control Appl Meth*. 2018;39:1833-1858 <https://doi.org/10.1002/oca.2445>

APPENDIX A

KERNEL AND BANDWIDTH SELECTION

As a measure of the discrepancy between $\hat{f}$ and f , one defines the mean integrated squared error (MISE)

$$\text{MISE} := \int (\hat{f}_{n,h}(x) - f(x))^2 dx.$$

Under smoothness and integrability assumptions on f (see the work of Sheather³³), we can define the main term in the Taylor expansion of the MISE as the asymptotic MISE (AMISE)

$$\text{AMISE} := \frac{1}{nh} \int K^2(x) dx + \frac{h^4}{4} \left(\int x^2 K(x) dx \right)^2 \int f''^2(x) dx, \quad (\text{A1})$$

which leads to the following choice for the bandwidth h minimizing (A1):

$$h_{\text{AMISE}} := \sqrt[5]{\frac{\int K^2(x) dx}{n \left(\int x^2 K(x) dx \right)^2 \int f''^2(x) dx}}. \quad (\text{A2})$$

If we substitute the optimal bandwidth given by (A2) in (A1), we obtain

$$\text{AMISE} = \frac{5}{4} \sqrt[5]{\frac{\left(\int K^2(x) dx \right)^4 \left(\int x^2 K(x) dx \right)^2 \int f''^2(x) dx}{n^4}}, \quad (\text{A3})$$

showing that the AMISE will tend to zero at a rate $n^{-4/5}$. Unfortunately though, the presence of the unknown factor $\int f''^2(x) dx$ in (A2) makes the expression of h_{AMISE} almost useless. For this reason, it might be more viable to approximate also the derivatives of f ^{35,41} or use one of the many practical ways^{31,33} for choosing the bandwidth using only information from the sample. Other results on the rate of convergence of the KDE have been proved in other works.⁴²⁻⁴⁵

On the practical side, a common choice for h , used in conjunction with the Gaussian kernel, is the SNR. Let S be the standard deviation of the sample $\{x_1, \dots, x_n\}$ of X , ie,

$$S := \sqrt{\frac{1}{n} \sum_{i=1}^n \left(X_i - \frac{\sum_{i=1}^n X_i}{n} \right)^2}.$$

The SNR bandwidth is then defined according to

$$h_{\text{SNR}} := 1.06 \frac{S}{\sqrt[5]{n}}. \quad (\text{A4})$$

Even though there is no general rule for obtaining an explicit value of h leading to the best approximation of f , it is important to point out that big values of h will probably lead to an overestimation of the volume of the density function and thus to a loss of information. As for the choice of the kernel, we want to show that, when using the AMISE expression for the approximation error, there is little room for improvement. In order to do this, we need to define the *efficiency* associated to a kernel. We first define the function

$$C(K) := \sqrt[5]{\left(\int x^2 K(x) dx \right)^2 \left(\int K^2(x) dx \right)^4}.$$

Substituting it into (A3), we have that minimizing (A3) with respect to K is equivalent to minimizing

$$\frac{5}{4} C(K) \sqrt[5]{\frac{\int f''^2(x) dx}{n^4}}.$$

This means that we should consider kernels with small values of $C(K)$. If we focus on kernels that are themselves PDFs (which are the only ones ensuring that the estimate is everywhere nonnegative), we have $\int K(x) dx = 1$. Moreover, we can also assume $\int x^2 K(x) dx = 1$. The fact that K is a density function guarantees that $\int x^2 K(x) dx$ is finite, thus allowing us to choose its normalized version in case $\int x^2 K(x) dx \neq 1$. Because our kernel satisfies

$$\int K(x) dx = \int x^2 K(x) dx = 1, \quad (\text{A5})$$

minimizing $C(K)$ reduces to minimizing $\int K^2(x) dx$, and in the work of Hodges and Lehmann,⁴⁶ it has been proven that the kernel

$$K_e(x) := \begin{cases} \frac{3}{4\sqrt{5}} \left(1 - \frac{1}{5}x \right) & |x| \leq \sqrt{5} \\ 0, & \text{else,} \end{cases} \quad (\text{A6})$$

TABLE A1 Efficiency of some kernels

Kernel	$K(\mathbf{x})$	eff(K)
Epanechnikov	$\begin{cases} \frac{3}{4\sqrt{5}} \left(1 - \frac{1}{5}x\right) & x \leq \sqrt{5} \\ 0, & \text{else} \end{cases}$	$\simeq 0.9939$
Biweight	$\begin{cases} \frac{15}{16}(1-x^2)^2 & x \leq 1 \\ 0, & \text{else} \end{cases}$	$\simeq 0.9859$
Triangular	$\begin{cases} 1 - x & x \leq 1 \\ 0, & \text{else} \end{cases}$	$\simeq 0.9295$
Gaussian	$\frac{1}{\sqrt{2\pi}} e^{-\frac{x^2}{2}}$	$\simeq 0.9512$
Rectangular	$\begin{cases} \frac{1}{2} & x \leq 1 \\ 0, & \text{else} \end{cases}$	$\simeq 0.9295$

achieves the minimal value of $C(K)$ under the constraints (A5). The efficiency of any kernel K satisfying (A5) is then defined as

$$\text{eff}(K) := \left(\frac{C(K_e)}{C(K)} \right)^{\frac{5}{4}} = \frac{3}{5 \int K^2(x) dx \sqrt{5 \int x^2 K(x) dx}},$$

where K_e is the Epanechnikov kernel, defined in (A6). Table A1 reports the efficiency of some of the most used kernels. Note that even the rectangular kernel (arguably the most naive choice of K) achieves an efficiency of $\simeq 0.93$. This leads us to the conclusion that, when measuring the error by means of (A1), the choice of the kernel is not as important as the choice of the bandwidth h .

APPENDIX B

MODEL OF FORCES FOR THE COMPLEX THREE-STAGE LAUNCHER

In Example 3 of Section 4, the launcher is subject to the force $\mathbf{F}_G$ due to gravity, the drag force $\mathbf{F}_D$, and the thrust force $\mathbf{F}_T$. The gravity force is

$$\mathbf{F}_G(m, \mathbf{x}) = - \begin{pmatrix} F_{Gx}(m, \mathbf{x}) & 0 & 0 \\ 0 & F_{Gy}(m, \mathbf{x}) & 0 \\ 0 & 0 & F_{Gz}(m, \mathbf{x}) \end{pmatrix} \frac{\mathbf{x}}{\|\mathbf{x}\|},$$

where

$$\begin{aligned} F_{Gx}(m, \mathbf{x}) &= F_{Gy}(m, \mathbf{x}) = m \frac{\mu_0}{\|\mathbf{x}\|^2} \left(1 + J_2 \frac{3}{2} \frac{R_e^2}{\|\mathbf{x}\|^2} \left(1 - 5 \frac{z^2}{\|\mathbf{x}\|^2} \right) \right), \\ F_{Gz}(m, \mathbf{x}) &= m \frac{\mu_0}{\|\mathbf{x}\|^2} \left(1 + J_2 \frac{3}{2} \frac{R_e^2}{\|\mathbf{x}\|^2} \left(3 - 5 \frac{z^2}{\|\mathbf{x}\|^2} \right) \right), \end{aligned}$$

and where μ_0 is the gravitation constant of the Earth (J_2 being the correction factor due to its oblateness). The drag force is

$$\mathbf{F}_D(\mathbf{x}, \mathbf{v}) = -F_D(\mathbf{x}, \mathbf{v}) \frac{\mathbf{v}_r(\mathbf{x}, \mathbf{v})}{\|\mathbf{v}_r(\mathbf{x}, \mathbf{v})\|},$$

where $F_D(\mathbf{x}, \mathbf{v}) = \frac{1}{2} \rho(\mathbf{x}) \|\mathbf{v}_r(\mathbf{x}, \mathbf{v})\|^2 A C_D(\mathbf{x}, \mathbf{v})$, ρ is the air density, and C_D is the drag coefficient, depending on the Mach number $M_a(\mathbf{x}, \mathbf{v}) = \frac{\|\mathbf{v}_r(\mathbf{x}, \mathbf{v})\|}{v_s(\mathbf{x})}$, which itself depends on the speed of sound v_s . The thrust force is

$$\mathbf{F}_T(\theta, \mathbf{x}, \mathbf{v}) = F_T(\mathbf{x}) \mathbf{i}_T(\theta, \mathbf{x}, \mathbf{v}),$$

where $F_T(\mathbf{x}) = g_0 \beta I_{sp} - SP(\mathbf{x})$, g_0 is the Earth gravitational acceleration, and P is the atmospheric pressure. The direction $\mathbf{i}_T$ is given by

$$\mathbf{i}_T(\theta, \mathbf{x}, \mathbf{v}) = \begin{cases} \frac{\mathbf{v}_r(\mathbf{x}, \mathbf{v})}{\|\mathbf{v}_r(\mathbf{x}, \mathbf{v})\|} & \alpha = 0, \\ \mathbf{R}_{\lambda_0, \phi_0} \mathbf{R}_\psi \mathbf{R}(\theta) \mathbf{e}_1 & \alpha \neq 0, \end{cases}$$

where

$$\begin{aligned} \mathbf{R}_{\lambda_0, \phi_0} &= \begin{pmatrix} -\sin(\lambda_0) & -\cos(\lambda_0) \sin(\phi_0) & \cos(\lambda_0) \cos(\phi_0) \\ \cos(\lambda_0) & -\sin(\lambda_0) \sin(\phi_0) & \sin(\lambda_0) \cos(\phi_0) \\ 0 & \cos(\phi_0) & \sin(\phi_0) \end{pmatrix}, \\ \mathbf{R}_\psi &= \begin{pmatrix} 0 & \sin(\psi) & -\cos(\psi) \\ 0 & \cos(\psi) & \sin(\psi) \\ 1 & 0 & 0 \end{pmatrix}, \\ \mathbf{R}(\theta) &= \begin{pmatrix} \cos(\theta) & -\sin(\theta) & 0 \\ \sin(\theta) & \cos(\theta) & 0 \\ 0 & 0 & 1 \end{pmatrix}, \\ \mathbf{e}_1 &= (1, 0, 0). \end{aligned}$$

The angles λ_0 and ϕ_0 are the longitude and the latitude of the launch site, and ψ is the launch azimuth.