Reliability Engineering and System Safety 134 (2015) 260–267

Contents lists available at ScienceDirect

Reliability Engineering and System Safety

journal homepage: www.elsevier.com/locate/ress

Sequential designs for sensitivity analysis of functional inputs in computer experiments

J. Fruth [a,*], O. Roustant [b], S. Kuhnt [a,c]

[a] Faculty of Statistics, TU Dortmund University, Vogelpothsweg 87, 44227 Dortmund, Germany
[b] Ecole Nationale Supérieure des Mines, FAYOL-EMSE, LSTI, F-42023 Saint-Etienne, France
[c] Dortmund University of Applied Sciences and Arts, Emil-Figge-Strasse 42, 44227 Dortmund, Germany

ARTICLE INFO

Available online 1 August 2014

Keywords:
Sensitivity analysis
Functional input
Sequential approach
Sheet metal forming
Springback

ABSTRACT

Computer experiments are nowadays commonly used to analyze industrial processes aiming at achieving a wanted outcome. Sensitivity analysis plays an important role in exploring the actual impact of adjustable parameters on the response variable. In this work we focus on sensitivity analysis of a scalar-valued output of a time-consuming computer code depending on scalar and functional input parameters. We investigate a sequential methodology, based on piecewise constant functions and sequential bifurcation, which is both economical and fully interpretable. The new approach is applied to a sheet metal forming problem in three sequential steps, resulting in new insights into the behavior of the forming process over time.

1. Introduction

In this work a general method is presented to study the influence of functional input and scalar parameters on a scalar output of a time-consuming computer model, where the interest lies in the behavior of the sensitivity over the functional domain. For illustration purposes, let us consider the particular application in sheet metal forming that originally motivated this research. Here a scalar output, springback, depends on two functional inputs: friction and blankholder force. The aim is to analyze the sensitivity of springback as a measure of geometrical accuracy to the input variables not only as a whole but also as functions over time corresponding to the different stages of the forming process. Let us look at a few preview runs, where only the friction between tool and metal is varied (Table 1). In the first two runs, the friction is constant over the 15 s of the punch travel. Then different functional input settings are used, indicating that varying friction can dramatically reduce springback. Although the four last runs have equal mean friction over time, we get very clear differences in the springback results. This clearly motivates the exploration of the functional behavior of the process, and more generally the sensitivity analysis of functional input parameters.

The most common method to study data with functional input is functional linear regression (see [1]), a framework to approximate, analyze and predict data with functional input. However, a design of experiments is not considered – data are assumed as already provided – and the interpretation of the influence of different time regions is not easy [2]. Furthermore, it is restricted to the assumption of linearity. Morris [3] introduced a Gaussian process model for the analysis of computer models with functional input and output, taking also the design of the functional experiments into account. However, he did not consider sensitivity analysis. Three ideas for the sensitivity analysis of computer experiments with one functional input are presented in Iooss and Ribatet [4]. They result in one uncertainty index for the input, and thus summarize the functional domain without giving an insight into the sensitivity with respect to changes at specific time intervals during the process.

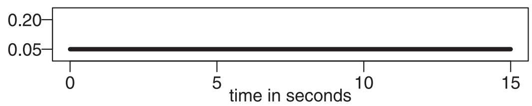
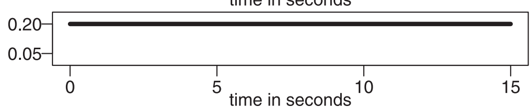
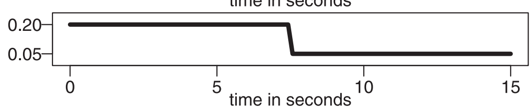
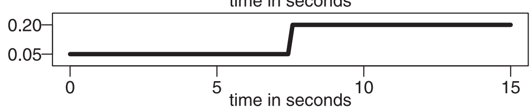
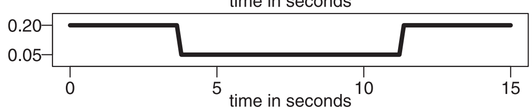
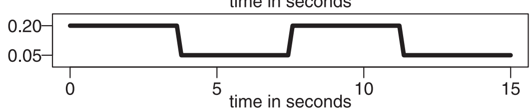
In this paper, sensitivity analysis of functional inputs is obtained by first transforming the functional problem into a scalar one by considering the basis of piecewise constant functions, often suggested in functional data analysis (see e.g. [1]). That framework also allows us to consider both scalar and functional inputs. Then we adapt the ideas of sequential bifurcation [5] in order to analyze more and more locally the effect of one specific part of the functional domain. For that purpose, we introduce a normalized sensitivity index, which allows comparison between two different steps, and investigate at a theoretical level its properties.

The paper is structured as follows. Sections 2 and 3 present the methodology, together with properties and a way for graphical

* Corresponding author.
E-mail addresses: fruth@statistik.tu-dortmund.de (J. Fruth), roustant@emse.fr (O. Roustant), sonja.kuhnt@fh-dortmund.de (S. Kuhnt).

http://dx.doi.org/10.1016/j.ress.2014.07.018
0951-8320/

Table 1
Preview runs to compare different functional settings of the parameter friction.

Functional input: friction (unitless)	Scalar output: springback (mm)
	6.1
	2.1
	14.7
	1.4
	0.8
	6.0

presentation. Section 4 describes the sequential design of the functional input. An application to sheet metal forming is given in Section 5, exploring the sensitivity of the time intervals of two functional inputs on springback. A discussion concludes the paper.

2. Framework

Our aim is to analyze an experiment involving d_s scalar input variables $x_i \in [-1,1]$, $i = 1, \dots, d_s$ and d_f functional input variables $g_j : D_j \mapsto [-1,1]$, $j = 1, \dots, d_f$ with a scalar output $Y \in \mathbb{R}$. For convenience the input parameters, scalar as well as functional, are bounded to fall into $[-1,1]$, and the input functions are defined on $D_j = [0,1]$ for each $j = 1, \dots, d_f$. The functions depend on a single argument, which in practical applications might often be the time, but the approach is not limited to it. The argument is not necessarily the same for all functional inputs. Beside the boundedness, there are no further conditions on the shape of the functions. Notice that this means that no assumption is made on the smoothness of the functions, i.e. that function values can vary sharply. Input and output parameters are connected by an unknown black-box function $f : [-1,1]^{d_s} \times \mathcal{F}_{[0,1]}^{d_f} \mapsto \mathbb{R}$:

$$Y = f(x_1, \dots, x_{d_s}, g_1, \dots, g_{d_f}),$$

where $\mathcal{F}_{[0,1]}$ denotes the space of all functions on [0, 1]. We assume this black-box function to be very expensive or time-consuming to execute.

Due to the functional inputs, the input domain is infinite dimensional. To cope with this issue, we propose to explore subintervals of the functional domains as a whole instead of looking at each point. The subintervals are then chosen sequentially to increasingly finer divisions. This approach is explained later in Section 4. For now, say we have a decomposition of the domain of each functional input g_j into $p_j \in \mathbb{N}^+$ subintervals at splitting points $\mathbf{a}_j = (a_j^0, \dots, a_j^{p_j})$ with $0 = a_j^0 < a_j^1 < \dots < a_j^{p_j - 1} < a_j^{p_j} = 1$,

$$D_j = [a_j^0, a_j^1[\cup [a_j^1, a_j^2[\ \cup \ \dots \ \cup \ [a_j^{p_j-1}, a_j^{p_j}].$$

We restrict each g_j to belong to $V_{\mathbf{a}_j}$, the space of piecewise constant functions over the subintervals defined by $\mathbf{a}_j$,

$$V_{\mathbf{a}_j} = \{Z_j^{(1)} \mathbb{1}_{[0,a_j^1[}(t) + \dots + Z_j^{(p_j)} \mathbb{1}_{[a_j^{p_j-1},1]}(t) \text{ with } Z_j^{(k)} \in [-1,1], 1 \le k \le p_j\}. \tag{1}$$

For a given set of splitting points $\mathbf{a}_j$ an element of $V_{\mathbf{a}_j}$ is then defined by the vector $(Z_j^{(1)}, \dots, Z_j^{(p_j)})$. Thus, we obtain a transformation of the input space of the black-box function from functional to scalar:

$$\begin{aligned} Y &= f(x_1, \dots, x_{d_s}, g_1, \dots, g_{d_f}) \\ &= \tilde{f}_{a^1, \dots, a^{p_{d_f}}}(x_1, \dots, x_{d_s}, Z_1^{(1)}, \dots, Z_1^{(p_1)}, \dots, Z_{d_f}^{(1)}, \dots, Z_{d_f}^{(p_{d_f})}). \end{aligned} \tag{2}$$

The described way to design the functional input can also be regarded in the context of function representation by basis functions (see e.g. [1, Chapter 3]). They can be viewed as B-splines of order one: linear combinations of piecewise constant functions being constant over one interval and zero otherwise [1,6]. They are also connected to wavelet theory, especially to Haar–Wavelets (see e.g. [7]), where the basic functions are piecewise constant functions, being scaled and shifted to the desired functional form. In this work we restrict ourselves to the basis of piecewise constant functions which allows a direct and easy interpretation. Nevertheless, we will borrow from wavelet theory the idea of splitting the domain sequentially. Hence, we will consider a sequence of embedded spaces $V_{\mathbf{a}_j} \subseteq V_{\mathbf{a}'_j} \subseteq V_{\mathbf{a}''_j} \subseteq \dots$ corresponding to refined time intervals.

A unified framework for scalar and functional inputs: Let us remark that the space of piecewise constant functions also includes scalar inputs, through the one-to-one correspondence:

$$\begin{aligned} &[-1,1] \to V_{(0,1)} \\ &x \to x \mathbb{1}_{[0,1]}(t) \end{aligned}$$

In other words any scalar input with value equal to x can be considered as a functional input with a constant value over [0, 1] equal to x. This allows us to consider only functional inputs, having in mind that scalar inputs correspond to functional inputs that are kept constant over [0, 1], and thus never split. We adopt this point of view from now on and denote by $d = d_s + d_f$ the total number of inputs.

3. Sensitivity indices

With the framework of Section 2, unifying the case of scalar and functional inputs, it is possible to perform sensitivity analysis on

$$Y = \tilde{f}_{a^1, \dots, a^{p_d}}(Z_1^{(1)}, \dots, Z_1^{(p_1)}, \dots, Z_d^{(1)}, \dots, Z_d^{(p_d)})$$

A specific sensitivity analysis method can be chosen according to the supposed behavior and evaluation costs of the experiment among methods such as regression analysis, Morris screening, Sobol indices or others (see e.g. [8] or [9] for an overview). Since the black-box function is assumed to be expensive, we have a very low budget. Thus, we suggest the method of regression analysis and use it in the further explorations. Nevertheless, the sequential design strategy proposed in Section 4 can be adapted to other methods.

Design of experiments for regression analysis usually covers only few levels of the input values, for instance two extreme values, encoded by -1 and $+1$. An equal scaling of the input