A Sequential Design Method for the Inversion of an Unknown System

Régis Bettinger, Pascal Duchêne, Luc Pronzato

To cite this version:

HAL Id: hal-00407826
https://hal.archives-ouvertes.fr/hal-00407826
Submitted on 27 Jul 2009

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

L’archive ouverte pluridisciplinaire HAL, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d’enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.
A Sequential Design Method for the Inversion of an Unknown System

Régis Bettinger**, Pascal Duchêne* Luc Pronzato**
* IFP-Solaize, Lyon, France, (e-mail: pascal.duchene@ifp.fr).
** Laboratoire I3S, Université de Nice Sophia-Antipolis/CNRS, France, (e-mail: bettinge@i3s.unice.fr, pronzato@i3s.unice.fr)

Abstract: A design method presented in a previous paper for the sequential generation of observation sites used for the inversion of a prediction model is extended to cope with practical issues such as delayed observations and design of batches of imposed size. The final objective of the construction is to be able to associate with any target \( T \) in the output space a value \( x_T \) of the input factors such that the response of the system at \( x_T \) will be “close” to \( T \) (from an industrial point of view, \( x_T \) corresponds to manufactory conditions that yield a product whose feature of interest is described by \( T \)). The problem is thus much different from the more standard one where one wishes to build a precise model over the whole input space: here the model only has to be precise over a set of values \( x_T \) that permit to reach any target \( T \), that is, the observation sites should not be spread over the entire admissible input space, but should rather concentrate in areas that cover the reachable output space when mapped by the system. Examples in low dimensions are presented that illustrate the behavior of the method and allow a comparison to be made with a standard sequential method for designing exploratory experiments.

Keywords: Kriging; system inversion; experimental design; sequential design; Tsallis entropy.

1. INTRODUCTION

We shall design an experiment (that is, choose observation sites \( X_1, \ldots, X_N \)) for a system
\[
S: \mathcal{X} \subset \mathbb{R}^m \rightarrow \mathbb{R}^p,
\]
for which no prior information is available. The input domain \( \mathcal{X} \) is a compact subset of \( \mathbb{R}^m \) and the reachable output domain \( \mathcal{Y} = S(\mathcal{X}) \subset \mathbb{R}^p \) is unknown. The objective of the design is to construct a model that will allow us to “invert” the system in the following sense: with any given reachable target vector \( T \in S(\mathcal{X}) \) one wishes to associate an input vector \( x_T \) such that the value \( S(x_T) \) is as close to \( T \) as possible.

The number \( N \) of observations to be made is fixed in advance and the experimental design problem consists in selecting \( N \) points \( X_1, \ldots, X_N \) \( \in \mathcal{X} \) such that the inverse problem above is solved with maximum accuracy after the observation of \( Y_i = Y(X_i), i = 1, \ldots, N \).

The input/output relationship is modeled by kriging (see, e.g., Sacks et al. [1989]; Santner et al. [2003]), which gives, at low computational cost, a prediction and an estimate of the prediction error at each point of the input domain (see Section 2). We use the Matlab toolbox DACE [Lophaven et al., 2002].

After having observed the output values \( Y^N = (Y_1, \ldots, Y_N) \top \) at the observation sites \( X_1, \ldots, X_N \) the posterior distribution \( P(Y(x)|Y^N) \) of the output at any point \( x \in \mathcal{X} \) is easily constructed. One can then define
\[
x_T = \arg\min_{x \in \mathcal{X}} \mathbb{E}\{||Y(x) - T||^2 | Y^N\} = \arg\min_{x \in \mathcal{X}} [||\hat{y}_N(x) - T||^2 + \hat{\sigma}_N^2(x)],
\]
the inverse prediction for \( T \), with \( \hat{y}_N(x) = \mathbb{E}\{Y(x) | Y^N\} \) and \( \hat{\sigma}_N^2(x) = \mathbb{E}\{||Y(x) - \hat{y}_N(x)||^2 | Y^N\} \) respectively the model prediction and variance at \( x \). A natural measure of the prediction accuracy for the target \( T \) is thus
\[
\zeta_N(T) = ||\hat{y}_N(x_T) - T||^2 + \hat{\sigma}_N^2(x_T),
\]
and two global criteria for the prediction accuracy over the whole reachable output space \( \mathcal{Y} \) can be considered,
\[
J_{M,N}(X_1, \ldots, X_N) = \max_{T \in \mathcal{Y}} \zeta_N(T),
J_{I,N}(X_1, \ldots, X_N) = \int_{\mathcal{Y}} \zeta_N(T) \mu(dT),
\]
with \( \mu(\cdot) \) some given probability measure on \( \mathcal{Y} \) indicating the relative interest among target values. A straightforward formulation of the design problem would consist in minimizing \( \mathbb{E}\{J_{M,N}(x_1, \ldots, x_N)\} \) or \( \mathbb{E}\{J_{I,N}(x_1, \ldots, x_N)\} \) with respect to \( x_1, \ldots, x_N \). This is, however, a formidable task and we need to follow another route (the quantities \( J_{M,N}(X_1, \ldots, X_N) \) and \( J_{I,N}(X_1, \ldots, X_N) \) cannot nevertheless be used to evaluate \textit{a posteriori} the quality of a given design in terms of inverse prediction performance, see Bettinger et al. [2008]).

Compared to [Bettinger et al., 2008], we consider a more natural extension to the case when responses are available with delay and several design points must be chosen simultaneously. The idea used in this previous paper consists in choosing observation sites that sample \( \mathcal{Y} \) uniformly as possible, in the sense that the \( N \) observed responses \( Y_i = Y(X_i), i = 1, \ldots, N \), are as spread as possible in \( \mathbb{R}^p \). Since this implies the exploration of the whole domain \( \mathcal{Y} \), we expect any possible target \( T \) in \( \mathcal{Y} \) to be “close” to a certain \( Y_i \). At the same time, one wishes that the \( X_i \)’s remain as concentrated as possible in \( \mathcal{X} \): indeed, model predictions must be precise for values \( x_T \) that
permit to reach any target $T$ and those inverse predictions $x_T$, obtained by (1), will preferably be chosen in areas where $\hat{\sigma}^2_n(x)$ is small, that is, in the neighborhood of some $X_i$’s. This concentration of design points in the input space can be obtained by constructing the design sequentially when the precision of the prediction is properly taken into account at each step, thereby favoring the choice of $X_{n+1}$ at step $n$ close to points $X_i$ already sampled. Choosing the point $X_{n+1}$ after $Y_1, \ldots, Y_n$ have been observed has also the advantage of using all the (increasing) information available and summarized in the current distributions $\mathcal{D}(Y(x)|Y^n)$, $x \in \mathcal{X}$. An initial design with $N_0$ points is chosen, with $N_0$ small compared to $N$. Since no prior information on the system is available, a space-filling design is used, ensuring that the first $N_0$ sites are as well spread in $\mathcal{X}$. As suggested by Morris and Mitchell [1995], we use a latin hypercube design, easy to generate, combined with the optimization of a maximin-distance criterion, see Section 4.2.

Two methods are proposed in [Bettinger et al., 2008] for choosing $X_{n+1}$:

- maximize the conditional (posterior) expectation of the minimum distance between $Y(x)$ and the $Y_i$’s already observed,

$$X_{n+1} = \arg \max_{x \in \mathcal{X}} \mathbb{E} \{ \min_{i=1, \ldots, n} \|Y(x) - Y_i\| | Y_1, \ldots, Y_n \},$$

see Section 3.2 (the idea being that, in the average sense, the next output $Y_{n+1}$ will be “far” from previous ones $Y_1, \ldots, Y_n$);

- maximize the expected second-order Tsallis entropy $H_2[\cdot]$ [Tsallis, 1988] of a kernel density estimator $\hat{\varrho}_{n,x}$ formed from the $Y_i$’s and $Y(x)$,

$$X_{n+1} = \arg \max_{x \in \mathcal{X}} \mathbb{E} \{ H_2[\hat{\varrho}_{n,x}] \},$$

see Section 3.3. The idea here is that maximum entropy over a compact set is obtained for the uniform distribution; we thus expect the method to spread the $Y_i$’s as much as possible in $\mathcal{Y}$ (the second-order Tsallis entropy being used because it allows very simple analytical calculations).

Note that the precision of the model prediction is taken into account by both approaches through the conditional distribution of $Y(x)$ used to evaluate the expectation $\mathbb{E} \{ \cdot \}$. In [Bettinger et al., 2008] we showed on a (noise-free) toy example with 2 inputs and 1 output that both approaches yield satisfactory designs.

For practical reasons, it may happen that batches of $k$ points $X_{n+1}, \ldots, X_{n+k}$ must be chosen simultaneously, and that those $k$ points must be selected before the $k$ previous observations are available (knowing, however, the input values $X_{n-k+1}, \ldots, X_n$ used in previous batch). This is the case in particular when the analysis of the outputs produced by the system is a time-consuming procedure and external constraints impose that the system $\mathcal{S}$ (experimental or simulated) is operated continuously.

A rather crude adaptation of the methods above to such a situation was proposed in [Bettinger et al., 2008] and illustrated by the application to a 5-inputs/2-outputs noisy system derived from real data collected in oil industry.

We show below that such practical constraints on delayed observations and batch design can easily be taken into account when using the design method based on the maximization of the expected entropy, without requiring any particular approximation (this is not the case, however, for the method based on the maximization of the expected minimum distance). The paper is organized as follows: Section 2 recalls the main properties of kriging predictors. Section 3 presents three sequential design methods, based on the maximization of the kriging variance $\hat{\sigma}^2_n(x)$, of the expected minimum distance and of the expected entropy, with their extensions to the “batch-delayed” case. In Section 4 the methods are compared on low-dimensional noise-free toy examples. The adaptation to the case of noisy observations, together with the need of specific optimization algorithms, are mentioned in a conclusion section.

2. PREDICTION BY KRIGING

For the sake of simplicity of the presentation we only consider the case of a single (scalar) response $Y(x)$. When several responses are present, we thus assume that they are modeled independently. Notice, however, that a more sophisticated approach (co-kriging) permits to take possible correlation between responses into account, see, e.g., Chilès and Delfiner [1999]. The model used for $Y(x)$ is

$$Y(x) = f^T(x) \beta + Z(x) + \varepsilon(x),$$

where $f(x) = (f_1(x), \ldots, f_k(x))^T$ is a vector of known regression functions, $\beta = (\beta_1, \ldots, \beta_n)^T$ is a vector of unknown parameters, $Z(\cdot)$ is a second-order stationary stochastic process with zero mean, and $\varepsilon(x)$ denotes the observation error at $x$. Those errors are assumed to be centered, independently distributed with variance $\sigma^2_\varepsilon$, and independent from $Z(x)$. We note $\text{Cov} \{ Z(x_1), Z(x_2) \} = \sigma^2_\rho(x_1 - x_2, \psi)$ the covariance function of $Z$, where $\sigma^2_\rho$ is the variance and $\rho(\cdot)$ the correlation function of $Z$, and $\psi$ is a vector of unknown parameters. Popular correlation functions include (with $\psi \in \mathbb{R}$, $t \in \mathbb{R}^m$)

$$\rho_c(t, \psi) = \prod_{i=1}^m e^{-|t_i|},$$

$$\rho_b(t, \psi) = e^{-\psi^2 |t|^2},$$

$$\rho_n(t, \psi) = \prod_{i=1}^m \frac{1}{\Gamma(\nu)2^{\nu-1}} \left( \frac{2\sqrt{\nu} |t_i|}{\psi} \right)^\nu K_\nu \left( \frac{2\sqrt{\nu} |t_i|}{\psi} \right),$$

called respectively the (isotropic) exponential, Gaussian and Matérn correlation functions, with $K_\nu$ the modified Bessel function of order $\nu$. The smoothness of the paths of $Z$ is related to the choice of the correlation function: in a nutshell, (2) corresponds to a process with non-smooth trajectories whereas (3) gives infinitely differentiable trajectories. In-between, the regularity of a model built with (4) grows with $\nu$.

Suppose one has collected $n$ observations $Y^n = (Y_1, \ldots, Y_n)^T$ at $X_1, \ldots, X_n$, denote $Z^n = (Z(X_1), \ldots, Z(X_n))^T$ and assume that $\sigma^2_\rho$ and the characteristics of the process $\rho(\cdot)$, $\sigma^2_\varepsilon$ and $\psi$ are known. The kriging predictor at an arbitrary point $x_0 \in \mathcal{X}$ is the best (in the sense of minimum variance) linear unbiased predictor at $x_0$. Easy calculations give
A classical approach for the sequential generation of design points in the presence of observations already performed. Using (7), we can compute the expected value of this quantity for an observation made at \( x \) and thus choose next design point as

\[ X_{n+1} = \arg \max_{x \in \mathcal{X}} E[d(Y(x), Y^n)] \mid Y^n \].

An analytic expression for \( E[d(Y(x), Y^n)] \mid Y^n \) is easily obtained when \( p = 1 \); for larger \( p \), however, the integrant has to be computed numerically, which becomes very time-consuming.

A generalization to the batch setting (\( k \geq 1 \)) could be based on

\[ (X_{n+1}, \ldots, X_{n+k}) = \arg \max_{X_{n} \in \mathcal{X}^k} \mathbb{E} \min_{i=1, \ldots, n+k, j=n+1, \ldots, n+k, j \neq i} \| Y_i - Y_j \| \mid Y^n \]
output space through entropy: since maximum entropy
over a compact set is obtained for the uniform distribution,
we maximize the (expected) entropy of the distribution of
the observations \{Y_1, \ldots, Y_n, Y(x)\} in \mathcal{Y}.

First, a kernel-density estimator of this distribution is
computed (see, e.g., Wand and Jones [1995])
\[
\hat{\phi}_{n,x}(y) = \frac{1}{n+1} \left[ \sum_{i=1}^{n} \varphi_{Y_i,h}(y) + \varphi_{Y(x),h}(y) \right] \tag{14}
\]
with \(\varphi_{z,h}(\cdot)\) the probability density function (p.d.f.) of
the normal \(\mathcal{N}(z,h^2)\). Usually, the smoothing parameter
\(h\) (window width) is taken decreasing with \(n\) in order
to ensure good asymptotic properties for the density estimator.
In the context considered here, however, the total number of observations \(N\) is small, and we keep \(h\)
constant (\(h = 0.01\), with the range of observed outputs
normalized to \([0, 1]\) at each step).

Second, the entropy of (14) is computed. A natural candi-
date is Shannon entropy, which, for \(\phi(\cdot)\) a p.d.f. on \(\mathbb{R}^p\),
can be written as \(H_1(\phi) = -\int_{\mathbb{R}^p} \phi(t) \log \phi(t) \, dt\). However,
this entropy functional does not yield an analytic formula
for a mixture of normal distributions such as (14). We
thus consider the second-order Tsallis entropy of the p.d.f.
\(\phi(\cdot)\), \(H_2(\phi) = 1 - \int_{\mathbb{R}^p} \phi^2(t) \, dt\), which can be given
an analytic expression when substituting the estimator (14)
for \(\phi\). Indeed, we have
\[
H_2[\hat{\phi}_{n,x}] = 1 - \frac{1}{(n+1)^2} \sum_{i=1}^{n+1} \varphi_{Y_i,h}(\varphi_{Y_i,h}) = \frac{1}{(n+1)^2} \sum_{i=1}^{n+1} \varphi_{Y_i,h}(\varphi_{Y_i,h}) - \frac{1}{n+1} \sum_{i=1}^{n+1} \varphi_{Y_i,h}(\varphi_{Y_i,h}) \tag{15}
\]
where we denote \(Y_{n+1} = Y(x)\). We thus obtain the
selection rule
\[
X_{n+1} = \arg \max_{x \in \mathcal{X}} \mathbb{E} \{ H_2[\hat{\phi}_{n,x}] | Y^n \}, \tag{16}
\]
with \(Y(x)\) having the distribution (7). One can notice that,
moreover, \(\mathbb{E} \{ H_2[\hat{\phi}_{n,x}] | Y^n \} \) can be given an
analytic expression using the property \(\int_{\mathbb{R}^p} \varphi_{a,\sigma}(t) \varphi_{b,\delta}(t) \, dt = \varphi_{a+b,\sqrt{\sigma^2 + \delta^2}}(b)\).

In the batch setting, we replace the kernel estimator (14)
by
\[
\hat{\phi}_{n,x}(y) = \frac{1}{n+k} \left[ \sum_{i=1}^{n} \varphi_{Y_i,h}(y) + \sum_{i=n+1}^{n+k} \varphi_{Y(x),h}(y) \right]
\]
and obtain the multiple-point selection rule
\[
(X_{n+1}, \ldots, X_{n+k}) = \arg \max_{X_{n+k} \in \mathcal{X}^k} \mathbb{E} \{ H_2[\hat{\phi}_{n,x}] | Y^n \}, \tag{17}
\]
which can still be given an analytic expression. In the
batch-delayed case, we use an approach similar to that
in Section 3.1 and choose \((X_{n+1}, \ldots, X_{n+k})\) as
\[
(X_{n+1}, \ldots, X_{n+k}) = \arg \max_{X_{n+k} \in \mathcal{X}^k} \mathbb{E} \{ H_2[\hat{\phi}_{n,x}] | Y^{n-k} \}, \tag{18}
\]

4. EXAMPLES

We first consider a 1-input/1-output example to illustrate
graphically the behaviors of selection rules (16, 17) and
(10, 11) and compare their performances. We then consider
the same 2-inputs/1-output example as in [Bettinger et al.,
2008] and generate the design with the rule (16), which
allows some comparison to be made with the case when
the points are added one-at-a-time with (15).

The kriging parameters are estimated by maximum likeli-
hood, excepted for \(\nu\) which is set to 3.7.

4.1 A toy example with \(m = p = 1\)

We illustrate the behaviors of the selection rules on a given
set of noise-free scalar observations depending on a scalar
input variable \(x \in \mathcal{X}\), with \(\mathcal{X}\) formed of 101 equally spaced
points in \([0, 1]\).

We first compare the influence of the correlation function
on the new set of points proposed by the design rule (16)
for batches of size \(k = 2\).

The locations of the points proposed by the algorithm is
plotted on Figure 1. The \(n = 6\) current observations (set
to arbitrary values) are represented by dots, the solid lines
correspond to the kriging predictions and the dashed lines
indicate the 95\% confidence intervals around the predictions.
The locations of the two design points generated by (16) are
indicated by stars. In the exponential case, the model is not
smooth and the confidence bounds are very loose; the new points
proposed by the design rule are close to previous points, in regions
where predictions are reliable. The fact that confidence bounds are
unreasonably tight for the Gaussian correlation is known in the kriging
literature, see [Stein 1999], and might produce an over-
confidence in the predictions, although this is not the case
here. Also, Gaussian correlation matrices are often poorly
conditioned [Ababou et al., 1994]; since inverses of correlation
matrices are required this raises numerical difficulties
(even if Cholesky factorization is used). Therefore, in the
rest of the paper we use (and in general recommend the use
of) the Matérn correlation function (4).

Consider now the behavior of the design rule (10) on the
same data set (with the Matérn correlation), see Figure 2.
Similarly to its well-known version (9) for \(k = 1\), the design
rule (10) enforces exploration and proposes points located
in low-sampled areas in the input space. It is therefore
not especially adapted to the model-inversion problem
considered in this paper.

In Figure 3 we compare the rules (17) and (11) with the
same set of observations as above (dots) and considering
the previous optimal batches of size 2 as being under
treatment (circles). One may notice that the entropy-based
rule (17) (top) places the next two points in areas where
predictions are precise and reasonably far from previous
observations (or expected observations for the two delayed
ones). The variance-based rule (11) (bottom) has the same
tendency as the rule (10) to generate new points far
from previous design points, thus enforcing a space-filling
property in the input domain. Although not adapted to
the context considered here, this rule can be of interest in
a batch-delayed setting when the objective is to build an
accurate model over the entire input domain.

4.2 A toy example with \(m = 2\) and \(p = 1\)

The system is now supposed to follow the equation (no
observation error, \(\sigma^2_x = 0\))
Fig. 1. Effect of the correlation function (exponential: top; Gaussian: middle; Matérn: bottom) on the choice of $k = 2$ new points with the rule (16).

\[
Y(x_1, x_2) = 0.2e^{x_1-x_2} + 2.2|x_2| + 1.3x_1^2 - 2x_2^2 - 0.5x_2^4 - 0.5x_2^4 + 2.5x_1^2 + 0.7x_1^3 + \sin(5x_1) \cos(3x_1^2) + \frac{3}{(8x_1 - 2)^2 + (5x_2 - 3)^2 + 1}.
\]

The initial design is a 9-points maximin latin hypercube (several such initial designs have been used, showing little influence on the performance); points are then added sequentially by pairs of $k = 2$ up to $N = 19$ with the rule (16). The input domain corresponds to a regular grid of $31 \times 31$ points in the square $[0,1]^2$. (This small number of points permits to obtain the optimal solution at each step by exhaustive search in a reasonable computing time. We noticed, however, that the results are rather sensitive to the grid size. Considering finer grids, with an optimization algorithm used at each step instead of exhaustive search, would thus be of interest.)

Figure 4 illustrates the way points are added at each step. The input values are plotted on the top, with dots representing the initial design, numbers indicating the sequence of batches generated and dashed lines corresponding to
contour plots of the response $Y(\cdot,\cdot)$. The associated output values are plotted on the bottom. Note that the points generated are concentrated in the input space; the dispersion of the outputs in $Y = [-0.67, 4.75]$ (bottom) is a bit worse than for $k = 1$, see Bettinger et al. [2008].

![Contour plots](image)

Fig. 4. Design points generated by (16) in a 2-inputs/1-output example: inputs (top) and outputs (bottom).

5. CONCLUSIONS

This paper proposed extensions of two sequential design rules to situations where runs should be made several-at-a-time and practical considerations impose a delay in the availability of observations. Simple examples illustrated that the first one is appropriate when the objective is to construct an accurate model over the whole input domain, while the second one is especially suited for model-inversion.

Both methods guarantee that there will be no repetition of observations at the same design site when there is no measurement error ($\sigma_r^2 = 0$). The situation is different when $\sigma_r^2 \neq 0$. In that case, we suggested in [Bettinger et al., 2008] to use the re-interpolation technique proposed by Forrester et al. [2006] for computer experiments. A first kriging predictor is constructed, taking the presence of measurement errors into account; it does not interpolate the data, that is, $\hat{g}_n(X_i) \neq Y_i$. A second kriging predictor is then constructed, assuming that there are no observation errors and using the predictions $\hat{g}_n(X_i)$ as if they were observations. One can show that for any $x$ the prediction at this second stage coincides with $\hat{g}_n(x)$ of the first stage, with the noticeable difference that the prediction variance for the second predictor is now zero at the observation points $X_i$, which permits to avoid repetitions.

For both approaches, generating $k$ points at a time requires the solution of a $k \times m$-dimensional optimization problem at each step, which quickly becomes computationally unfeasible when $k$ increases (for the low dimensional examples considered, with $m = 1, 2$ and $k = 2$, the input domain corresponded to a regular grid and all possible choices of design points could be tested at each step). The development of suitable algorithms (e.g., of the exchange type) is under current investigation and forms a prerequisite to the solution of the 5-inputs/2-outputs model-inversion problem presented in [Bettinger et al., 2008].

REFERENCES


