2015

Dynamic calibration of drifting sensor arrays for real-time monitoring

Zongyu Geng

DYNAMIC CALIBRATION OF DRIFTING SENSOR ARRAYS FOR REAL-TIME MONITORING

Zongyu Geng

Dissertation submitted to the The Statler College of Engineering and Mineral Resources
at West Virginia University
in partial fulfillment of the requirements
for the degree of

Doctor of Philosophy
in
Industrial Engineering

Dr. Feng Yang, Committee Chairperson
Dr. Nianqiang Wu
Dr. Majid Jaridi
Dr. Wafik Iskander
Dr. Natalia A. Schmid

Industrial and Management Systems Engineering

Morgantown, West Virginia
2015

Keywords: Sensor calibration, transient sensor signals, sensor drift, forward and inverse calibration, multivariate Gaussian processes, transfer function regression

Abstract

DYNAMIC CALIBRATION OF DRIFTING SENSOR ARRAYS FOR REAL-TIME MONITORING

Zongyu Geng

Traditional sensor calibration is restricted to mathematically relating the steady-state sensor responses to the target analyte concentrations to realize environment monitoring. However, commonly-used chemical sensors usually require a relatively long time, on the order of minutes, to reach steady-state operation, and exhibit nonlinear drifting behaviors. To achieve real-time monitoring of rapidly-changing environments while accommodating drifting behaviors, this work develops statistical methods for both forward descriptive calibration and inverse dynamic calibration of sensor arrays.

Forward calibration is performed based on experimental data. In this work, multivariate Gaussian processes (GPs) were adapted to obtain the forward calibration model, which quantifies the sensor response as a function of the analyte concentration, the drifting variables, and the sensors' exposure time. The multivariate GP method synergistically models all calibration data collected under a range of drifting conditions, and seeks to produce the calibration model of highest quality with the given experimental data. The forward calibration model is a descriptive model, relating sensors' time-dependent responses to a static environment specified by several variables, hence it is not able to assist in real-time monitoring of rapidly-changing environments.

To achieve real-time monitoring of analyte concentrations while fully utilizing the efficiency of forward calibration rendered by multivariate GPs, an inverse calibration method was developed. This inverse model takes the form of a transfer function regression, infers the time-varying analyte concentrations from the dynamic sensor responses, and thus can be coupled with sensors for real-time monitoring. The inverse transfer function model is estimated from the pseudo-calibration data generated by the forward multivariate GP model, which captures the sensors' dynamic and drifting behaviors as reflected in the real experimental data.

Simulated sensor arrays have been developed from real sensor data, and were used to demonstrate the calibration methods developed in this work.

ACKNOWLEDGMENTS

I would first like to sincerely thank Dr. Feng Yang, my advisor, for her guidance and patience as we worked through my research. She was always there ready to help me with any difficulties. She was always extremely supportive to me. I would like to thank her again for her help during my transition time which was the most difficult time in my life.

I would also like to thank Dr. Majid Jaridi and Dr. Wafik Iskander for their wonderful course teaching. They were always extremely supportive to me.

I am also thankful to Dr. Nianqiang Wu and Dr. Natalia A. Schmid for their inspiring ideas whenever we reached the dead end in our research there.

And last but not least, I would like to thank my family. They have been a great inspiration for me during all my years in school. My wife sacrificed a lot for me to support my study and research. My parents have given me all the love and support with the discipline that I need for my life and I really hope I can make them proud.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Sensor arrays are widely used to quantify analytes of interest in applications such as manufacturing control, environment monitoring, security, biomedicine, food industry, etc [2]. A sensor array consists of a number of sensors whose responses can be used as a "fingerprint" for the target analytes in an environment. An individual sensor is subject to cross-sensitivity, which causes it to respond to multiple background analytes. Thus it is clear that the calibration modeling technique is an essential component of any sensor array, which provides a mathematical function relating the analyte concentrations to the array responses. This work focuses on i) how to determine the best functional form of the calibration model and ii) how to estimate it with the highest quality.

In many applications, the concentrations of target analytes are likely to change rapidly. For instance, in combustion control sensors are integrated with feedback controllers to realize fine tuning of the operating condition [3, 4]. To achieve real-time monitoring, sensors must be able to keep up with the pace of the time-varying analyte concentrations. Conventional sensor calibration develops mathematical models associating steady-state sensor responses to the target analyte concentration. Within the scope of steady-state calibration, a sensor is required to be highly responsive (i.e., be able to reach its steady state quickly) in order to provide real-time monitoring. However, the commonly used electrochemical sensors or semiconductor-based resistive sensors usually show

delayed responses to the rapid changes of analyte concentrations, due to the limitation of materials and the underlying sensing mechanisms. As illustrated in Figure 1.1, when the actual physical change happens in an environment, several minutes are needed to restore equilibrium; that is, to complete multiple steps in the sensing signal generation: diffusion of the analyte to the sensor surface, adsorption of the analyte on the sensor surface, chemical interaction of the analyte with the sensor, and signal transduction. Unfortunately, it is almost impossible to physically improve the response-time performance to the extent that is required for real-time monitoring [5]. In light of this, an appealing alternative is provided by dynamic calibration: modeling a sensor's dynamic behavior so that its transient signals, as opposed to steady-state responses, can be used to track the analyte concentration.

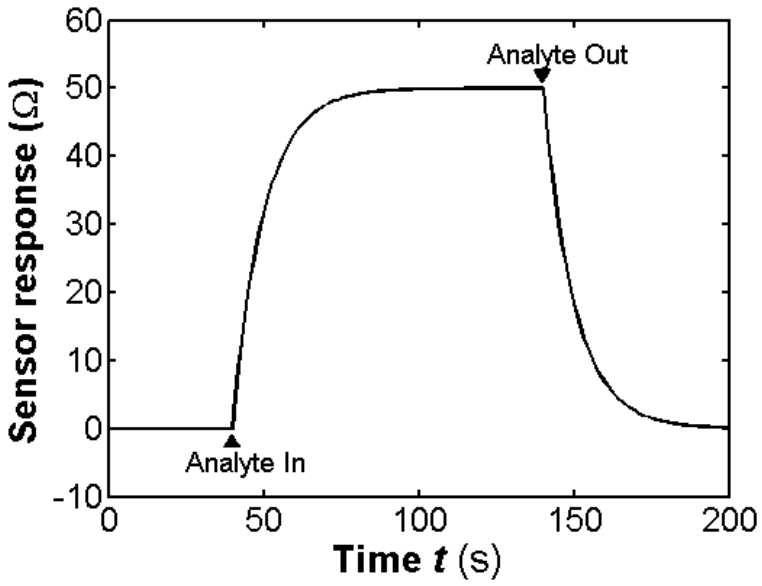


Figure 1.1 Time-dependent responses of a chemical sensor.

A sensor's steady-state and dynamic behavior is subject to drifts, which can be caused by physical and chemical changes of the sensor over time, and/or through repeated use. Examples of such changes include thermomechanical fatigue, re-organization of the sensing material, and irreversible interaction with aggressive analytes [6–9]. A drifting sensor will respond differently

when exposed to the same environment, as illustrated in Figure 1.2. Stages 1, 2 and 3 represent the three stages of time or usage across which the sensor's behavior has drifted. As can be seen in Figure 1.2, the response curve may change in both responding time and sensitivity. A drifting sensor needs to be re-calibrated before it can be used again. It is time-consuming and costly to perform sufficient experiments for full re-calibration of a drifting sensor array in each stage; hence, it is of significant practical interest to minimize the additional experimental effort needed at each stage to achieve an adequate calibration model for that stage.

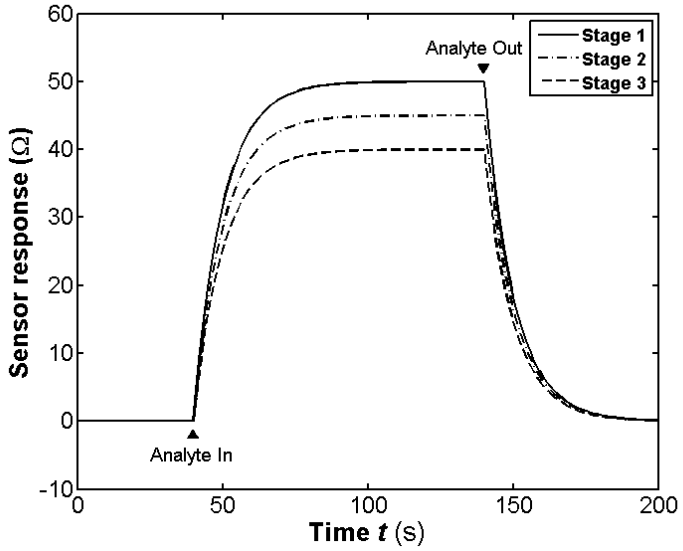


Figure 1.2 An example of sensor drift.

The ultimate objective of this work is to enable a drifting sensor array to monitor the rapid changes in the concentrations of multiple analytes in real time. There are two major challenges. First, multiple variables such as the concentrations, exposure time and usage stage (for drifting effects), may affect the array responses. They do so in a multivariate nonlinear fashion, and also interact nonlinearly with each other. Second, for real-time monitoring, the calibration model needs to allow for the inference of unknown analyte concentrations from observed sensor responses. The statistical model family selected for sensor calibration needs to satisfy two basic requirements:

the model family needs to be sufficiently flexible and powerful to capture the comprehensive and nonlinear behavior of the sensor, and it must be possible to derive valid statistical inferences, based on the model fitting, to quantify the uncertainty of the model's estimates.

In this work, statistical methods were developed to estimate both the forward and inverse calibration models. The forward model takes the form of a multivariate GP, which is highly flexible and able to capture any continuous multivariate functional relationships. The forward multivariate GP models the sensor responses as a function of the analyte concentrations, and exposure time in a static environment, and of the usage-stage variable. The inverse model takes the form of a transfer function regression, and models the analyte concentrations as a function of observed dynamic sensor responses for a certain usage stage. The inverse model is fitted from the pseudo-calibration data generated by the forward model, which efficiently models the real experimental data to capture the sensor array's dynamic and drifting behaviors. This is the first known attempt to efficiently calibrate the dynamic and drifting behaviors of sensor arrays for real-time monitoring of rapidly-changing environments.

The remainder of this thesis is organized as follows. In Chapter 2, the existing work relevant to this work was reviewed. An overview of our proposed methods can be found in Chapter 3. Based on the multivariate GP model, a new statistical calibration procedure is detailed in Chapter 4 to calibrate the dynamic and drifting performance of a sensor array. In Chapter 4, we also discuss the estimation of the inverse calibration model based on transfer function regressions. Empirical results are provided in Chapter 5 via a simulated chemical sensor array. Conclusion follows in Chapter 6.

Chapter 2

Literature Review

2.1 Transient Analysis of Sensor Signals

Considering that the transient period in a typical chemical sensor response is on the order of a minute, several modeling attempts (e.g., [10–12]), since the late 1990s, have studied transient sensor signals. What is particularly relevant to this research is the modeling work that seeks to functionally relate the transient sensor responses to the target analyte concentration [13–17]. The majority [13–16] proposed the utilization of transient features in addition to steady-state responses to make inferences about the analyte of interest. The underlying idea is that the transient signals may provide valid information about the analyte that steady-state responses cannot, and would increase the quantification accuracy when the steady-state information is available. Apparently, it does not improve the real-time performance of sensors which require a long time to reach steady-state operation.

To the best of our knowledge, the only exceptional work is Muezzinoglu et al. [17], which adopted a moving average operator to extract features from sensor transients, and solely utilized the extracted transient features for analyte quantification. Although their transient features are

correlated with the analyte concentration, and are available much earlier than the steady-state features, the major limitation of Muezzinoglu's method lies in its applicability to static environments. The amount of target analyte is assumed to be unchanging over time in the environment of interest, such that a "clear" transient feature can be extracted. This assumption is not valid when the analyte concentration is also time-varying, as in most field use.

2.2 Drift Reduction Methods

Conventional methods use reference-based linear regression or linear compensation methods to quantify drifting effects [7,18]. Recognizing the possible nonlinear nature of sensor drift, powerful nonlinear models such as neural networks [19, 20] and kernel ridge regressions [21, 22] have also been employed. For instance, Hossein et al. [19] proposed modeling the fluctuations of a chemical sensor under different environmental factors, based on an Artificial Neural Network (ANN) model. The fitted ANN takes the sensor response, temperature and humidity as the inputs, and calculates the actual analyte concentration as the output. However, in this stream of nonlinear modeling work, no effort was ever made to quantify the uncertainties of the target estimates (e.g., the analyte concentration estimated by the calibration model from an observed sensor response). This is at least partly due to the difficulties in deriving valid statistical inferences (i.e., quantifying model uncertainties) based on those models [23, 24].

Multivariate approaches have been developed based on sensor arrays [25–27]. Sensor arrays have become a major focus of research because they are more robust against instrumental noises and sensor drift and because, in the presence of multiple target analytes, it is difficult to develop an individual sensor which is highly sensitive to all analytes. A sensor array can be built to mitigate this constraint by utilizing responses from multiple less-sensitive sensors.

Very little work has been done to reduce the drift in time-dependent sensor response, with the

notable exception of Wenzel et al. [28]. They proposed the utilization of the Kalman Filter formalism to estimate both analyte concentration and the amount of sensor drift in real time. However, there are two major limitations of their method. First, they did not look into the calibration methods, and assume the "true" sensor dynamics are known *a priori*. Second, they only consider the additive drifting term and omit other multiplicative drifting effects, such as changes in sensitivity and response time.

2.3 Gaussian Process Modeling

The Gaussian process (GP) model, also known as the Kriging model, is an interpolation model widely used in many areas, including spatial statistics [29, 30] and computer experiments [31]. Empirical works have shown its superiority over other interpolating techniques such as splines [32]. The GP model interpolates data at the observation point, and the variance becomes larger as the prediction point moves away from the observation points [1].

To understand the basic use of Gaussian Process model, consider the following simple example of a regression problem. Let x be the independent predictor, and y the dependent response variable. A finite set of observation data is (x_i, y_i), $i = 1, 2, 3, \ldots$, which are depicted as "+" in Figure 2.1. Over the horizon-Axis, $y(\cdot)$ is assumed to be a Gaussian process distribution, i.e. the correlation between any two responses $y(x)$ and $y(x')$ is a function of their predictors (x, x'). With fitted model parameters, the expected value $\widehat{y}(\cdot)$ can be plotted as the solid curve in Figure 2.1. It is clear that the advantage of GP models is the flexibility and capability of capturing any continuous functional relationship. Another advantage of GP models is the availability of valid inferencing methods. For instance, the 95% confidence region of the expected response can be calculated, as the shaded area represents in Figure 2.1.

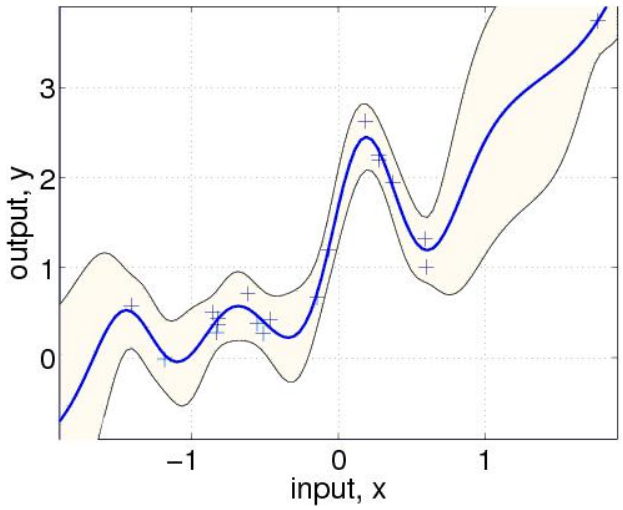


Figure 2.1 An example of Gaussian Process regression [1].

The characteristics of a GP model is determined by its covariance structure

$$\mathrm{COV}(y(x), y(x')) = \tau^2 K(x, x'), \tag{2.1}$$

with τ^2 the constant variance and $K(x,x')$ the correlation function, which is non-negative definite in the sense that for every pair x and x', $K(x,x') \geq 0$. If the process is stationary, $K(x,x')$ depends on their separation, $x - x'$, while if non-stationary it depends on the actual position of the points x and x'. The selection of correlation functions is discussed in [1]. There exist several approach to estimating all these unknown parameters. Oakley J [33] developed an analytical expression for the variance term. Ankenman et al. [34] proposed a step-by-step procedure to estimating all these unknown parameters in the correlation function, plus the variance of random observation errors. Lehman [35] compared several methods to estimate the unknown parameters in these models and recommended the maximum likelihood estimator (MLE). Heuristic algorithms can be used to iteratively get the MLEs for unknown parameters in correlation functions simultaneously.

The GP's availability of valid inferences has motivated research work in the design of experiments [31, 36–38]. Built on GP's inference ability, Geng et al. [38] developed an experimental

design method to achieve efficient sampling of calibration data in a batch sequential manner. The resulting calibration procedure, which integrates the GP-based modeling and experimental design, was applied to a simulated chemiresistor sensor to demonstrate its effectiveness and efficiency over the traditional method.

To analyze regression problems with multiple responses, it is necessary to apply the multivariate GP model, which assumes that cross-correlation exists between different types of responses. However, most of the GP or Kriging literature ignores multivariate cases, often reducing these multiple responses to a univariate response, or combining all responses via a weighting function. In multivariate GPs, a major problem remains in selecting the covariance function which assures that the covariance matrix of all responses will be positive definite.

A natural generalization of the covariance (2.1) is the separable model [39, 40], whose covariance matrix separated into two components: (i) between-responses covariance matrix, accounts for the cross-correlations between different response types; (ii) a correlation function over the independent variables (i.e. inputs), accounts only for the auto-correlations between responses of the same type for different input combinations. With the second component, it is implied that all response variables have the same auto-correlation matrix. An advantage of the separable covariance approach is mathematical tractability. However, all these approaches suffer from the disadvantage that the correlation function of each of the responses are identical. This assumption is often not justified in many applications.

Alternatively, the nonseparable models are based on either convolution method or the linear model of coregionalization (LMC). The convolution method is introduced to this community by Boyle and Frean [41]. Examples include that of Majumdar and Gelfand [42], who observed that the convolution of two positive-definite covariance functions is again positive definite. However, Majumdar and Gelfand noted that the convolution will have no closed form in covariance functions, and can only be handled numerically. This causes great inconvenience in practice use, i.e.

high requirements in computation and storage. Hankin [43] proposed a non-separable covariance structure, which is a more general extension of Fricker et al. [44]'s method, named the linear model of co-regionalization (LMC). Hankin's model allows different correlation in multivariate Gaussian Processes, and meanwhile provides a closed-form covariance matrix, based on which valid model fitting strategy is available and has been developed in our work.

Chapter 3

Problem Statement and Method Overview

For the convenience of discussion, we give in Table 3.1 notations of variables used in our calibration procedure. The task of array calibration herein includes two parts: (i) providing a dynamic inverse calibration model, by which the time-varying concentrations $\mathbf{c}(t)$ can be inferred from the observed array responses $\mathbf{r}(t)$ at a certain usage stage; (ii) providing a forward calibration model describing the dynamic and drifting behavior of sensor arrays at various usage stages.

Table 3.1 Variables in sensor calibration

Variables	Description
$t = 0, 1, 2, \ldots$	the discrete index of exposure time.
$\mathbf{c} = (c_1, c_2, \ldots, c_P)$	the concentration vector of the P analytes in an environment
$\mathbf{r} = (r_1, r_2, \ldots, r_Q)$	the array responses Q components. (It is assumed $P = Q$)
$s = 1, 2, \ldots, S$	sensors' usage stage (or age), across which drifting has occured. Within each stage, it is assumed that no significant drift occurs.

3.1 Inverse Dynamic Calibration

For a given usage stage, the inverse calibration model models the time-varying analyte concentrations $\mathbf{c}(t)$ as a function of the dynamic sensor responses $\mathbf{r}(t)$. However, conventional sensor calibration develops mathematical models associating only the steady-state sensor responses to the static analyte concentrations. In this work, we developed a dynamic calibration model which utilizes transient sensor responses to estimate the underlying analyte concentrations which may change rapidly.

The general form of the inverse calibration model can be written as

$$\mathbf{c}(t) = \mathbf{G}(\mathbf{r}(t), \mathbf{r}(t-1), \mathbf{r}(t-2), \ldots, \mathbf{r}(t-B)) \tag{3.1}$$

The positive integer B represents the time order of the model. In general, the time order B barely exceeds two [45], and hence we allow B to be up to three. In the empirical demonstration (Chapter 5), it has shown that a dynamic inverse model enables the estimation of $\mathbf{c}(t)$ from observed transient responses $\mathbf{r}(t)$ within several seconds, whereas the time for the sensor to completely reach steady state is in the order of minutes.

In this work, the dynamic inverse model is estimated from the pseudo data generated by the forward calibration model, which is briefed in Section 3.2.

3.2 Gaussian Process-Based Forward Calibration

The purpose of the forward calibration is to accurately describe the dynamic and drifting behavior of a sensor array based on a least additional experimental effort for the drifted array. Suppose that the sensor array has drifted to enter its usage stage s^*. The forward calibration seeks to quantify the dependence of $\mathbf{r}(t)$ upon $c(t)$ and s ($s = 1, 2, \ldots, s^*$) based on all the experimental data collected from the stages $1, 2, \ldots, s^*$.

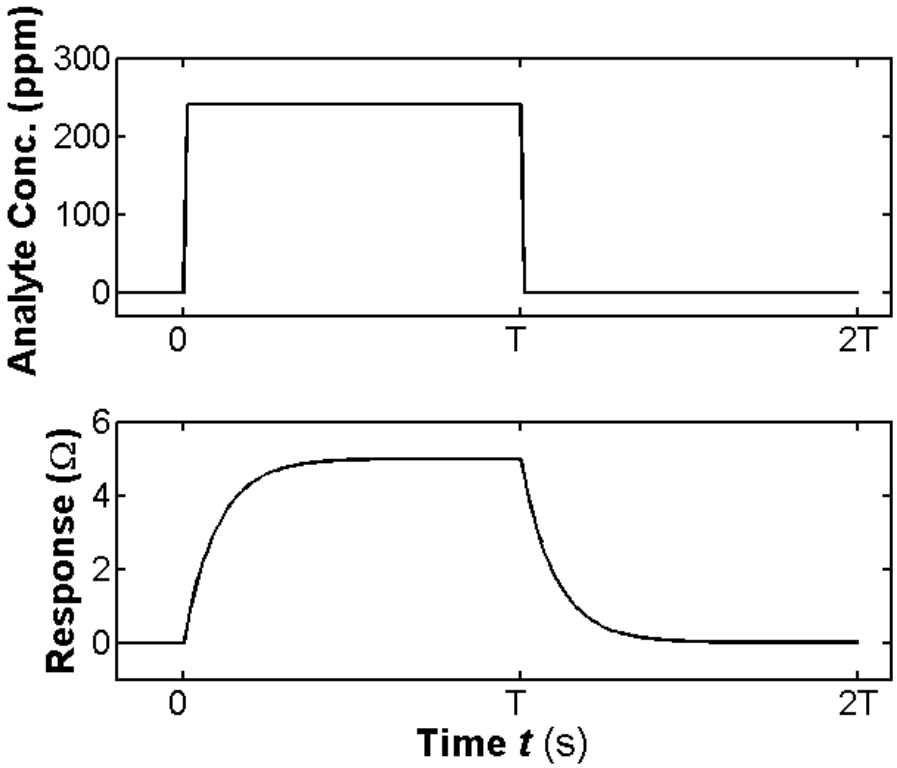


Figure 3.1 An example of the exposure cycle (from $t = 0$ to $t = 2T$).

In this work, a multivariate GP model is adapted to describe the array responses in an exposure cycle as depicted in Figure 3.1. The model can be written in general as:

$$\mathbf{r} = \mathrm{E}(\mathbf{r}(\mathbf{w})) + \boldsymbol{\varepsilon} = \mathbf{F}(\mathbf{w}) + \boldsymbol{\varepsilon} = \mathbf{F}(\mathbf{c}, t, s) + \boldsymbol{\varepsilon}. \tag{3.2}$$

The time index t denotes the exposure time of sensors (Figure 3.1); each component of $\mathbf{c} = (c_1, c_2, \ldots, c_P)$ represents the peak concentration of an analyte in an exposure cycle (Figure 3.1); the random error vector $\boldsymbol{\varepsilon} = (\varepsilon_1, \varepsilon_2, \ldots, \varepsilon_Q)$ is assumed to be independent and identically (i.i.d.) distributed; the expected response $\mathrm{E}(\mathbf{r}(\mathbf{w}))$ is modeled by the vector function $\mathbf{F}(\cdot)$, a multivariate Gaussian Process.

It is clear that the two calibration models (3.1) and (3.2) take different modeling directions. Following the classic literature of sensor calibration [46, 47], we consider $\mathbf{c} \rightarrow \mathbf{r}$ as the forward direction, obtaining $\mathbf{r}$ for a given $\mathbf{c}$; and the inverse direction refers to $\mathbf{r} \rightarrow \mathbf{c}$, estimating $\mathbf{c}$ for a

given $\mathbf{r}$. Apparently, the inverse model (3.1) can be used directly for real-time quantification of the analytes, which is of more interest to the practical use of sensors. But it can not reflect the nature of sensors' performance, especially when drift occurs. Besides, with the predictor variable $\mathbf{r}$ being random and the response $\mathbf{c}$ deterministic, standard statistical inference methods cannot be applied. On the other hand, the forward model (3.2) can be used to fully describe the sensors' behavior, but does not allow for a direction calculation of $\mathbf{r} \rightarrow \mathbf{c}$.

In light of the pros and cons of the forward and inverse models, we propose to approach the statistical modeling issues as follows:

- The forward model is developed based on multivariate GP describing the dynamic and drifting behavior of sensor arrays in an exposure cycle across usage stages.
- At the current stage $s = s^*$, the inverse model (3.1) is used for real-time monitoring due to its practical convenience. The model parameters are estimated from the "pseudo" data, which are generated from the forward model accurately describing the sensor array's behavior.

3.3 Overview of the Calibration Procedure

The array calibration is carried out off-line, and outlined in Figure 3.2. The procedure takes as an input the sensor array of interest, whose usage stages are $s = 1, 2, \ldots, s^*$ with s^* being the current stage. As drift occurs, its forward calibration model functionally relates the sensor response $\mathbf{r}(t)$ to $\mathbf{w} = (\mathbf{c}, t, s)$. In Step (1) Figure 3.2, calibration experiments will be performed on the stage-s sensor following the cycle pattern (Figure 3.1). In Step (2) Figure 3.2, the multivariate GP model is adapted, and the methodology details are discussed in Section 4.1. All the experimental data obtained so far from stages $s = 1, \ldots, s^*$ are utilized to obtain the fitted forward calibration model, which will be used to generate the "pseudo" data as described in Step (3). Finally, the inverse dynamic model is estimated in Step (4), and employed to assist the sensor array for real-time

monitoring in its operational use.

Input: The sensor array with usage stage $s = s^*$, where s^* represents the current stage.

(1) At each stage, cycled experiments (Figure 3.1) have been performed leading to the calibration data $(\mathbf{r}, \mathbf{w})$ for stages $s = 1, 2, \ldots, s^*$.

(2) Based on all the experimental data collected so far, estimate the multivariate GP-based forward calibration model

$$\mathbf{r} = \widehat{\mathbf{F}}(\mathbf{w}) + \widehat{\boldsymbol{\varepsilon}} \tag{3.3}$$

(3) Set the exposure condition $\mathbf{w}_0 = (\mathbf{c}, t, s = s^*)^\top$ to obtain the predictive results $\widehat{F}(\mathbf{w}_0)$ as the pseudo data. The pseudo data includes a number of exposure cycles. Each of them can be denoted as pairs of $\{\mathbf{c}(t), \mathbf{r}(t); t = 0, 1, 2, \ldots, 2T\}$.)

(4) Based on the pseudo data, estimate the inverse dynamic model:

$$\mathbf{c}(t) = \widehat{\mathbf{G}}(\mathbf{r}(t), \mathbf{r}(t-1), \mathbf{r}(t-2), \ldots) \tag{3.4}$$

Output: The fitted inverse dynamic model (3.4) for stage $s = s^*$.

Figure 3.2 Forward Calibration

The algorithm in Figure 3.3 will be carried out to achieve real-time monitoring of the analytes' concentration. When a stage-s^* sensor is exposed to an unknown environment, the algorithm allows the analyte concentration to vary continuously with time. The related computation can be completed within seconds (e.g. on a computer with Pentium 4 CPU and 2G RAM, it takes less than one second).

Given: (i) The fitted inverse dynamic calibration model (3.4) for the target sensor array at stage $s = s^*$;

(ii) The most recently observed responses $\{\mathbf{r}(t^*), \mathbf{r}(t^*-1), \mathbf{r}(t^*-2), \ldots\}$, with t^* as the current time point we stand.

Do Plug $\{\mathbf{r}(t^*), \mathbf{r}(t^*-1), \mathbf{r}(t^*-2), \ldots\}$ into the fitted model (3.4) to calculate $\mathbf{c}(t^*)$.

Output: The estimated concentration $\widehat{\mathbf{c}}(t^*)$ at t^*.

Figure 3.3 Real-time monitoring via the inverse calibration model.

Chapter 4

Methods

Two types of calibration models are developed in this work. The multivariate GP-based forward calibration model is fitted from all the experimental data across multiple usage stages of sensors, and describes the sensor array's dynamic and drifting behavior (detailed in Section 4.1). The inverse dynamic calibration model for the array's current usage stage is fitted from the pseudo data generated by the forward model, and can be integrated with the sensing device for real-time monitoring of rapidly-changing environments (detailed in Section 4.2).

4.1 Forward Calibration via Multivariate Gaussian Process (GP)

As mentioned in Section 3.2, an MPG is employed in this work to model the functional dependence of $\mathbf{r}$ upon $\mathbf{w} = (\mathbf{c}, t, s)$. Let Q be the number of individual sensors in a array, and $H = P + 2$ the dimension of the exposure condition vector $\mathbf{w}$. The forward calibration model (3.2) is represented as

$$\mathbf{r} = \mathbf{F}(\mathbf{w}) + \boldsymbol{\varepsilon} = \boldsymbol{\mu} + \mathsf{M}(\mathbf{w}) + \boldsymbol{\varepsilon}, \tag{4.1}$$

where $\boldsymbol{\mu}$ is the mean vector, and $\mathsf{M}(\mathbf{w})$ is a realization of a mean-zero stationary Q-dimensional GP with the constant $Q \times Q$ variance matrix Ψ. The error vector $\boldsymbol{\varepsilon} = (\varepsilon_1, \varepsilon_2, \ldots, \varepsilon_Q)^\top$ follows a continuous multivariate probability distribution (e.g., Normal distribution). In this work, it is assumed that the variance of $\boldsymbol{\varepsilon}$ is a constant matrix $\Sigma_{\boldsymbol{\varepsilon}}$ independent of $\mathbf{w}$, which is a typical assumption in sensor calibration [48, 49]. However, it is worthy of noting that multivariate GP allows for general variance structures of errors, as shown in Ankenman et al. [34].

A multivariate GP is characterized by its correlation structure [1]. For given $\mathbf{w}$ and $\mathbf{w}'$, the correlation function between the u^{th} and the v^{th} sensor responses ($1 \leq u \leq v \leq Q$) is denoted as

$$K_{(u,v)}(\mathbf{w}, \mathbf{w}') \in [-1, 1]. \tag{4.2}$$

$$\Sigma_{\mathsf{M}}(\mathbf{w}, \mathbf{w}') = \begin{pmatrix} \Psi_{11}K_{(1,1)}(\mathbf{w}, \mathbf{w}') & \Psi_{12}K_{(1,2)}(\mathbf{w}, \mathbf{w}') & \ldots & \Psi_{1Q}K_{(1,Q)}(\mathbf{w}, \mathbf{w}') \\ \Psi_{21}K_{(2,1)}(\mathbf{w}, \mathbf{w}') & \Psi_{22}K_{(2,2)}(\mathbf{w}, \mathbf{w}') & \ldots & \Psi_{2Q}K_{(2,Q)}(\mathbf{w}, \mathbf{w}') \\ \vdots & \vdots & \ddots & \vdots \\ \Psi_{Q1}K_{(Q,1)}(\mathbf{w}, \mathbf{w}') & \Psi_{Q2}K_{(Q,2)}(\mathbf{w}, \mathbf{w}') & \ldots & \Psi_{QQ}K_{(Q,Q)}(\mathbf{w}, \mathbf{w}') \end{pmatrix}. \tag{4.3}$$

where Ψ denotes the constant variance matrix of the Q univariate GP processes.

As to the specific form of the correlation function $K_{(u,v)}(\cdot, \cdot)$, traditional multivariate GP methods [39, 40] assume $K_{(u,v)}(\cdot, \cdot)$ to be identical for any (u, v) pair, and thus, the variance-covariance matrix can be separated as $\Sigma_{\mathsf{M}}(\mathbf{w}, \mathbf{w}') = \Psi \cdot K(\mathbf{w}, \mathbf{w}')$, which is referred to as a "separable" covariance structure. Apparently, the assumption of identical correlations is only valid if any pair of univariate GPs shares the same correlation structure. In this work, we adopt the general non-separable covariance structure proposed by Hankin [43], and the form of the correlation function can be written as:

$$K_{(u,v)}(\mathbf{w}, \mathbf{w}') = \exp\left\{ \sum_{h=1}^{H} \frac{-(\frac{1}{2}\boldsymbol{\Theta}_{u;h}^{-1} + \frac{1}{2}\boldsymbol{\Theta}_{v;h}^{-1})^{-1}(w_h - w_h')^2}{|(\frac{1}{2}\boldsymbol{\Theta}_{u;h} + \frac{1}{2}\boldsymbol{\Theta}_{v;h})(\frac{1}{2}\boldsymbol{\Theta}_{u;h}^{-1} + \frac{1}{2}\boldsymbol{\Theta}_{v;h}^{-1})|^{\frac{1}{4}}}, \right\} \tag{4.4}$$

where $\mathbf{\Theta}$ is a $Q \times H$ unknown parameter matrix. The function (4.4) allows for different correlations across different (u, v) pairs, and is proved to lead to a positive-definite variance matrix $\Sigma_{\mathsf{M}}(\mathbf{w}, \mathbf{w}')$. Compared to other convolution-based non-separable covariance structure, the correlation (4.4) has the advantage of computational convenience, and statistically valid estimation methods (Section 4.1.2). It is worth mentioning that given $Q = 1$, the correlation function (4.4) can be reduced to the squared exponential form, which is widely used in univariate GP models. [1].

For a collection of I distinct settings $\{\mathbf{w}_i; i = 1, 2, \ldots, I\}$ with I being a positive integer, a stacked vector can be defined as $(\mathsf{M}(\mathbf{w}_i)^\top, \mathsf{M}(\mathbf{w}_i)^\top, \ldots, \mathsf{M}(\mathbf{w}_i)^\top)^\top$ with QI elements, and the corresponding $QI \times QI$ variance matrix is

$$\mathbf{M}(\Psi, \mathbf{\Theta}) = \begin{pmatrix} \Psi & \Sigma_{\mathsf{M}}(\mathbf{w}_1, \mathbf{w}_2) & \ldots & \Sigma_{\mathsf{M}}(\mathbf{w}_1, \mathbf{w}_I) \\ \Sigma_{\mathsf{M}}(\mathbf{w}_2, \mathbf{w}_1) & \Psi & \ldots & \Sigma_{\mathsf{M}}(\mathbf{w}_2, \mathbf{w}_I) \\ \vdots & \vdots & \ddots & \vdots \\ \Sigma_{\mathsf{M}}(\mathbf{w}_I, \mathbf{w}_1) & \Sigma_{\mathsf{M}}(\mathbf{w}_I, \mathbf{w}_2) & \ldots & \Psi \end{pmatrix}. \tag{4.5}$$

Along with an arbitrary condition $\mathbf{w}_0$, we further define the following $Q \times QI$ matrix

$$\mathbf{N}(\mathbf{w}_0; \Psi, \mathbf{\Theta}) = \begin{pmatrix} \Sigma_{\mathsf{M}}(\mathbf{w}_0, \mathbf{w}_1), \Sigma_{\mathsf{M}}(\mathbf{w}_0, \mathbf{w}_2), \ldots, \Sigma_{\mathsf{M}}(\mathbf{w}_0, \mathbf{w}_I) \end{pmatrix}. \tag{4.6}$$

4.1.1 Experimental Data for Forward Calibration

To calibrate a sensor array, experimental data need to be collected following the exposure cycle ($t = 0, 1, 2, \ldots, 2T$ as in Figure 3.1) at a range of different concentrations and usage stages. The calibration sample data can be represented as

$$\{(\mathbf{w}_i, \mathbf{r}_j(\mathbf{w}_i)); i = 1, 2, \ldots, I, \; j = 1, 2, \ldots, n\}. \tag{4.7}$$

In (4.7), I is the number of distinct experimental conditions $\mathbf{w}$; $\mathbf{r}_j(\mathbf{w}_i) = (r_{j1}, r_{j2}, \ldots, r_{jQ})^\top$ denotes the $Q\mathbf{y}1$ response vector from the j^{th} replication at $\mathbf{w}_i$, and n is the number of replications performed. The sample average at $\mathbf{w}_i$ can then be calculated as

$$\bar{\mathbf{r}}(\mathbf{w}_i) = \frac{1}{n}\sum_{j=1}^{n} \mathbf{r}_j(\mathbf{w}_i), \tag{4.8}$$

and

$$\bar{\mathsf{R}} = (\bar{\mathbf{r}}(\mathbf{w}_1)^\top, \bar{\mathbf{r}}(\mathbf{w}_2)^\top, \ldots, \bar{\mathbf{r}}(\mathbf{w}_I)^\top)^\top, \tag{4.9}$$

is the $QI \times 1$ stacked vector of sample averages at the I distinct settings.

4.1.2 Model Fitting

Given the experimental data (4.7), in which the sensor responses are subject to random errors, the unknown model parameters $(\Sigma_{\boldsymbol{\varepsilon}}, \boldsymbol{\mu}, \Psi, \boldsymbol{\Theta})$ can be obtained by following the steps in Figure 4.1. To solve the high-dimensional nonlinear optimization problem (4.11), a genetic algorithm (GA) search method is adopted [50].

Given the real calibration data collected $\{(\mathbf{w}_i, r_j(\mathbf{w}_i)); i = 1,2,\ldots,I,\ j = 1,2,\ldots,n\}$, we denote $\mathbf{e}_{ij}$ as the j^{th} residual at the design point $\mathbf{w}_i$, which is calculated as

$$\mathbf{e}_{ij} = \mathbf{r}_j(\mathbf{w}_i) - \widehat{\mathbf{r}}(\mathbf{w}_i), \tag{4.12}$$

which is later used in our bootstrapping algorithm to quantify the estimation uncertainty of inferred concentration values (Section 4.2.2).

Algorithm multivariate GP-Fitting

Input: The sample data $\{(\mathbf{w}_i, \mathbf{r}_j(\mathbf{w}_i)); i = 1,2,\ldots,I,\ j = 1,2,\ldots,n\}$.

Process:

(i) Based on the sample data, calculate the sample averages $\bar{\mathsf{R}}$, as given in (4.9).

(ii) Estimate the variance matrix $\widehat{\Sigma}_{\boldsymbol{\varepsilon}}$ of the random error $\boldsymbol{\varepsilon}$ as

$$\widehat{\Sigma}_{\boldsymbol{\varepsilon}(u,v)} = \frac{1}{I(n-1)} \sum_{i=1}^{I} \sum_{j=1}^{n} (r_{ju}(\mathbf{w}_i) - \bar{r}_u(\mathbf{w}_i))(r_{jv}(\mathbf{w}_i) - \bar{r}_v(\mathbf{w}_i)). \tag{4.10}$$

(iii) Obtain $(\widehat{\boldsymbol{\mu}}, \widehat{\Psi}, \widehat{\boldsymbol{\Theta}})$ by maximizing the log-likelihood function:

$$\begin{aligned} \mathcal{L}(\widehat{\boldsymbol{\mu}}, \widehat{\Psi}, \widehat{\boldsymbol{\Theta}}) = &- \ln[(2\pi)^{QI/2}] - \frac{1}{2} \ln \left| \mathbf{M}(\Psi, \boldsymbol{\Theta}) + (\mathbf{1}_I \mathbf{1}_I{}^\top) \otimes \widehat{\Sigma}_{\boldsymbol{\varepsilon}} \right| \\ &- \frac{1}{2} (\bar{\mathsf{R}} - \mathbf{A}\widehat{\boldsymbol{\mu}})^\top \left[\mathbf{M}(\Psi, \boldsymbol{\Theta}) + (\mathbf{1}_I \mathbf{1}_I{}^\top) \otimes \widehat{\Sigma}_{\boldsymbol{\varepsilon}} \right]^{-1} (\bar{\mathsf{R}} - \mathbf{A}\widehat{\boldsymbol{\mu}}), \end{aligned} \tag{4.11}$$

with $\mathbf{A} = \mathbf{1}_I \otimes \mathcal{I}_Q$. Let $\mathcal{I}_Q$ be the identity matrix of size Q, and $\mathbf{1}_I$ an $(I \times 1)$ vector of ones.

Output: Estimated model parameters $(\widehat{\Sigma}_{\boldsymbol{\varepsilon}}, \widehat{\boldsymbol{\mu}}, \widehat{\Psi}, \widehat{\boldsymbol{\Theta}})$.

Figure 4.1 The fitting algorithm of the multivariate Gaussian process (GP) model for forward calibration.

4.1.3 Forward Estimation Based on Multivariate Gaussian Processes

With the fitted parameters $(\widehat{\Sigma}_{\boldsymbol{\varepsilon}}, \widehat{\boldsymbol{\mu}}, \widehat{\Psi}, \widehat{\boldsymbol{\Theta}})$ for the multivariate GP calibration model, the expected sensor responses at an arbitrary setting $\mathbf{w}_0$ can be estimated as:

$$\begin{aligned} \widehat{\mathbf{r}}(\mathbf{w}_0) &= \widehat{\mathbf{F}}(\mathbf{w}_0) \\ &= \widehat{\boldsymbol{\mu}} + \mathbf{N}(\mathbf{w}_0; \Psi, \boldsymbol{\Theta}) \left[\mathbf{M}(\Psi, \boldsymbol{\Theta}) + (\mathbf{1}_I \mathbf{1}_I{}^\top) \otimes \widehat{\Sigma}_{\boldsymbol{\varepsilon}} \right]^{-1} (\bar{\mathsf{R}} - \mathbf{A}\widehat{\boldsymbol{\mu}}), \end{aligned} \tag{4.13}$$

with $\mathbf{A} = \mathbf{1}_I \otimes \mathcal{I}_Q$. Let $\mathcal{I}_Q$ be the identity matrix of size Q, and $\mathbf{1}_I$ an $(I \times 1)$ vector of ones.

As explained in Section (3.2), the forward calibration model $\widehat{\mathbf{F}}$ will be used to generate the "pseudo" data for the inverse calibration of the sensor array in the current usage stage. The forward model makes a most efficient use of all the experimental data across usage stages, and can be used "for free" to generate as much pseudo data as needed to generate a high-quality inverse model.

4.2 Inverse Dynamic Calibration

When a sensor array is exposed to an unknown environment, the inverse calibration model is used to estimate the concentrations of the multiple target analytes from the observed array responses. As we discussed in previous chapters, both the concentrations and sensor responses are dynamically changing with time.

The inverse calibration model

$$\mathbf{c}(t) = \mathbf{G}(\mathbf{r}(t), \mathbf{r}(t-1), \ldots, \mathbf{r}(t-B)). \tag{4.14}$$

takes the form of a Transfer Function (TF) model. Originally introduced for describing dynamic behavior of stochastic systems [45], TF models (mainly linear TF models) have been used in signal modeling for estimating damped sinusoids and exponentials [51–53]. In this work, we are using the TF modeling techniques, which, to the best of our knowledge, is the first attempt to use TF models (possibly nonlinear TF models) to fully calibrate the dynamic profile of a sensor array, and to assist the real-time monitoring of evolving environments.

Given a usage stage s^*, the fitted forward calibration model will be used to generate the "pseudo" sample data set, from which the inverse model (4.14) will be estimated. By using the bootstrapping technique, we can also quantify the estimation uncertainty of inferred concentration values. We next discuss the two parts in Section 4.2.1 and Section 4.2.2 respectively.

4.2.1 Transfer Function Model

The inverse calibration model (4.14) is used to track the time-varying analyte concentrations for real-time monitoring. It takes as input sensor responses, which change with time and may never reach steady state, and estimates the underlying analytes' concentrations, which may be time-varying. In this work, the functional form of (4.14) is assumed to be a polynomial model with polynomial order being L. That is, $\mathbf{c}(t)$ is approximated as a model that may incorporate up to L^{th} order terms of each individual variable included in the set $\{\mathbf{r}(t), \mathbf{r}(t-1), \ldots, \mathbf{r}(t-B)\}$ and their interaction effects [54]. For instance, a full quadratic model ($1 \leq p \leq P; 1 \leq q \leq Q$) is given as

$$c_p(t) = \sum_{q_1=1}^{Q-2} \sum_{q_2=q_1+1}^{Q-1} \sum_{b_1=0}^{B-1} \sum_{b_2=b_1+1}^{B} \sum_{i=1}^{L-2} \sum_{j=i+1}^{L-1} h_{pq_1q_2ijb_1b_2} [r_{q_1}(t-b_1)]^i [r_{q_2}(t-b_2)]^j + \sum_{q=0}^{Q} \sum_{b=0}^{B} \sum_{i=0}^{L} h_{pqqiibb} [r_q(t-b)]^i. \tag{4.15}$$

where all the coefficients are unknown parameters.

We adopt the polynomial functional form (4.15), which incorporates not only linear (first-order) but also nonlinear (higher-order) terms, out of the following considerations. (i) If the steady-state concentration-response relationship for a sensor deviates somewhat from linearity [55], then a linear TF model will be insufficient to describe that sensor's transient behavior. (ii) In our empirical experience, a nonlinear TF model provides a more accurate estimation of analyte concentrations than its linear counterpart, judging from cross validation-based evaluation.

For a given usage stage, the functional terms actually included in the final model will be determined from the pseudo data set $\{\mathbf{c}(t), \mathbf{r}(t)\}$, through a stepwise model selection procedure [54], which aims at finding the estimated model of the simplest functional form but adequate to describe the sensor dynamics. We start with the simplest model form with a constant term only, and then seek to expand the model by including higher-order (time order and/or polynomial order) functional terms that make a significant contribution in terms of describing the dynamic relationship.

As new terms are incorporated, the existing functional terms may be removed from the model if they no longer play a significant part. The procedure is terminated when no more functional terms are eligible for inclusion or removal. During the stepwise selection, the estimation of a candidate model is performed using the least-square methods [54]. Once the inverse model has been obtained for a sensor array, it can be used to timely estimate the analytes' concentration through simple recursive computations.

4.2.2 Estimation Uncertainty of Analyte Concentrations

In operational use of a sensor array, the uncertainty of $\widehat{\mathbf{c}}(t)$, which is measured by $\mathrm{Var}[\widehat{\mathbf{c}}(t)]$, is also of great interest to end users. $\mathrm{Var}[\widehat{\mathbf{c}}(t)]$ stems from two sources. The first source lies in the random errors to which the sensor responses $\mathbf{r}$ are subject. Since $\widehat{\mathbf{c}}(t)$ is a function of $\mathbf{r}$, $\mathrm{Var}[\widehat{\mathbf{c}}(t)]$ is affected by $\mathrm{Var}[\mathbf{r}]$; The second source comes from the uncertainty of the model $\widehat{\mathbf{G}}$, which is fitted from the sample data, either experimental or "pseudo". Unfortunately, analytical expressions are not available for the uncertainty of $\widehat{\mathbf{c}}(t)$ [49, 56]. Hence, we adapted a bootstrapping method [57–61] to evaluate $\mathrm{Var}[\widehat{\mathbf{c}}(t)]$ through resampling (Figure 4.2).

Under the assumption that the error terms in the calibration model (4.1) are i.i.d., the pool of residuals $\{\mathbf{e}_{ij};\, i = 1,2,\ldots,I,\; j = 1,2,\ldots,n\}$ are employed in the bootstrap algorithm (Figure 4.2). Bootstrap resampling is performed in both Step 1 and 3 of this algorithm, following the forward direction as the real sampling does. The fitted GP calibration model $\widehat{\mathbf{F}}$ and the resampled residuals are used to simulate sensor responses mimicking real responses. The resampling in Step 1 accounts for the uncertainty of the fitted multivariate GP model, while the resampling in Step 3 simulates the randomness of the observed sensor response. Following the recommendations in the bootstrapping literature [62, Chap. 17, page 572] [57, Chap. 6, page 50], we set the resampling size N_{B1} as 50 in Step 1, and N_{B2} as 25 in Step 3. Both sources of uncertainty are reflected in the bootstrapping outputs $\{\widehat{\mathbf{c}}^{*}_{0,b,k};\, b = 1,2,\ldots,N_{B1}, k = 1,2,\ldots,N_{B2}\}$, which can be used to estimate $\mathrm{Var}[\widehat{\mathbf{c}}_0(t)]$ as

shown in (4.16), Figure 4.2.

For given $t \in [0, 2T]$, $\widehat{\text{Var}}[\widehat{\mathbf{c}}_0(t)]$ is a $P \times P$ matrix, with a diagonal element represented as $(\widehat{\text{Var}}[\widehat{\mathbf{c}}_0(t)])_{p,p}$ ($p = 1, 2, \ldots, P$). The standard error (SE)

$$\text{SE}[\widehat{c}_{0p}(t)] = \sqrt{(\widehat{\text{Var}}[\widehat{\mathbf{c}}_0(t)])_{p,p}} \tag{4.17}$$

measures the uncertainty of $\widehat{c}_{0p}(t)$, the estimated concentration of the p^{th} component in the sample mixture at the time point t. In operational use, the analyte concentrations may vary continuously, and does not follow the cycle pattern in FIgure 3.1. To be more general, we let $\text{SE}[\widehat{c}_{0p}]$ be the maximum value of $\text{SE}[\widehat{c}_{0p}(t)]$ over the exposure cycle $t = 0, 1, 2, \ldots, 2T$, such as

$$\text{SE}[\widehat{c}_{0p}] = \max_{t=0}^{2T}(\text{SE}[\widehat{c}_{0p}(t)]). \tag{4.18}$$

The SE (4.18) can be used to form an approximate interval estimate for the c_{0p} in real-time monitoring, regardless of the sampling time:

$$[\ \widehat{c}_{0p} - z_{1-\alpha/2} \cdot \text{SE}[\widehat{c}_{0p}],\ \widehat{c}_{0p} + z_{1-\alpha/2} \cdot \text{SE}[\widehat{c}_{0p}]\ ] \tag{4.19}$$

where α is a small percentage which typically takes values between the range $[0.05, 0.1]$, and $z_{1-\alpha/2}$ is the $100(1-\alpha/2)^{th}$ percentile of the standard normal distribution [54]. For $\alpha = 0.05$, $z_{1-\alpha/2} = 1.96$. The interval (4.19) can be interpreted as follows: The unknown true component concentration c_{0p} falls within this interval with a chance of $1 - \alpha$.

Algorithm GP-Bootstrapping

Inputs: (a) a given exposure condition of interest $\mathbf{w}_0 = (\mathbf{c}_0, t, s^*)^\top$ with $0 \le t \le 2T$; (b) $\{\mathbf{w}_i; i = 1,2,\ldots,I\}$, the exposure conditions in the pseudo data; (c) $\widehat{\mathbf{F}}(\mathbf{w})$, the multivariate GP model fitted from the experimental data; (d) $\{\mathbf{e}_{ij};\ i = 1,2,\ldots,I, j = 1,2,\ldots,n\}$, the pool of residuals obtained from (4.12); (e) N_{B1}, the resampling size for calibration data; and N_{B2}, the resampling size for observed sensor response.

Process:

Repeat Step 1–3 for $b = 1,2,\ldots,N_{B1}$

1. For each design point $\{\mathbf{w}_i; i = 1,2,\ldots,I\}$,

- Set $\mathbf{w}^*_{i,b} = \mathbf{w}_i$.
- Generate $\mathbf{r}_j(\mathbf{w}^*_{i,b}) = \widehat{\mathbf{F}}(\mathbf{w}^*_{i,b}) + \mathbf{e}^*_{ij,b}$, where $j = 1,2,\ldots,n$, and $\mathbf{e}^*_{ij,b}$ is randomly selected from the residual pool with replacement.

2. Based on the bootstrap data $\{(\mathbf{w}^*_{i,b}, \mathbf{r}_j(\mathbf{w}^*_{i,b})); i = 1,2,\ldots,I\}$, fit the inverse calibration model (4.14), which is denoted as $\widehat{\mathbf{G}}^*_b(\cdot)$.

3. Given $\mathbf{w}_0 = (\mathbf{c}_0, t, s^*)^\top$, repeat Step (i)–(ii) for $k = 1,2,\ldots,N_{B1}$

(i) Generate $\mathbf{r}^*_{k,0} = \widehat{\mathbf{F}}(\mathbf{w}_0) + \mathbf{e}^*_k$, with e^*_k randomly picked from the residual pool with replacement.

(ii) Estimate the analyte concentrations $\widehat{\mathbf{c}}^*_{0,b,k}(t) = \widehat{\mathbf{G}}_b(\mathbf{r}^*_{k,0})$.

Outputs: The bootstrap estimates of $\widehat{\mathbf{c}}_0(t)$: $\{\widehat{\mathbf{c}}^*_{0,b,k}(t); b = 1,2,\ldots,N_{B1}, k = 1,2,\ldots,N_{B2}\}$, from which $\mathrm{Var}[\widehat{\mathbf{c}}_0(t)]$ can be estimated as:

$$\widehat{\mathrm{Var}}[\widehat{\mathbf{c}}_0(t)] = (N_{B1} \times N_{B2})^{-1} \sum_{b=1}^{N_{B1}} \sum_{k=1}^{N_{B2}} (\widehat{\mathbf{c}}_0(t) - \mathbf{c}_0(t))^2. \tag{4.16}$$

Figure 4.2 The Gaussian process-based bootstrapping algorithm.

Chapter 5

Empirical Results

Our proposed method was applied to calibrate a chemiresistor sensor array, whose responses substantially drift over long terms of use. The effectiveness of the calibration procedure is demonstrated.

5.1 The Simulated Chemiresistor Sensors

The evaluation of a statistical method such as our calibration procedure requires an extremely large amount of validation data (as will be exemplified in Section 5.2 and 5.3), and is usually performed based on simulation, as opposed to real experiments, which could be prohibitively expensive. Built from real data, simulation models are able to capture any important features of real data [63], and at the same time can be used to generate data via computer experiments. Hence, in this work, a simulated sensor array was developed to evaluate the proposed procedure.

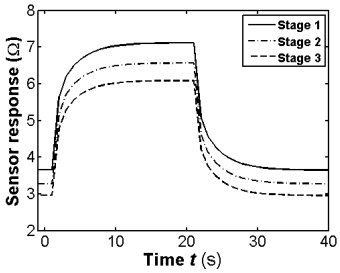
The simulation models (Appendix A) are derived from the experimental data of real chemiresistors given in Vergara et al. [27]. With continuous measurement on a sensor array over three years, Vergara's data is the most comprehensive one in the stream of sensor drift study. The simulated sensor array consists of three sensors ($Q = 3$), and aims at quantifying the time-varying

concentrations from a mixture of three target analytes ($P = 3$). Generally, it is required that $Q \geq P$ since otherwise $\mathbf{c}$ cannot be identified even from an error free $\mathbf{r}$ [47] [64, Chap. 4]. Our simulation model (Figure A.1) takes inputs as the usage stage s and analyte concentrations $\{\mathbf{c}(t) = (c_1(t), c_2(t), c_3(t)); t = 0, 1, 2, 3, \ldots\}$, with $t = 0$ being the initial time point. The corresponding outputs are the expected vector of sensor responses $\{\mathrm{E}[\mathbf{r}(\mathbf{c}, t, s)]; t = 1, 2, 3, \ldots\}$. The functional expression can be found in Section A.2. It takes the form of a multiple exponential decay function [10] to represent the time-dependent response curves, and is estimated from the real calibration data provided by Vergara et al. [27]. Figure 5.1 plots as examples the exposure cycles ($2T = 40$ seconds) of each simulated sensor over three usage stages. We assume that at $t = 0$, no analyte is present and sensors hold their steady-state. Thus, the initial responses $\mathbf{r}(t = 0)$ is the baseline signal. The analyte concentrations will step-wisely change to given levels at $t = 1$, and back to zero at $t = T + 1$. Figure 5.1 displays the following features: (i) Over the three usage stages, each sensor drifts in both the baseline response and the sensitivity; (ii) each sensor is subject to cross-sensitivity; (iii) and each sensor targets one particular analyte in the sense that its response to this analyte is much more pronounced than the other sensors' responses, which comply with the array design guidelines in Geng et al. [65].

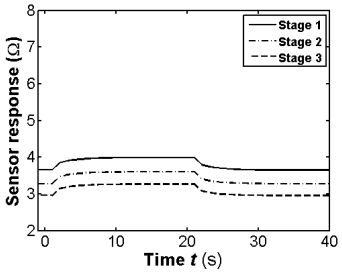
For any given inputs, the sensor responses are simulated as follows to mimic real sensor signals subject to random errors:

$$\mathbf{r}(\mathbf{c}, t, s) = \mathrm{E}[\mathbf{r}(\mathbf{c}, t, s)] + \boldsymbol{\varepsilon} \tag{5.1}$$

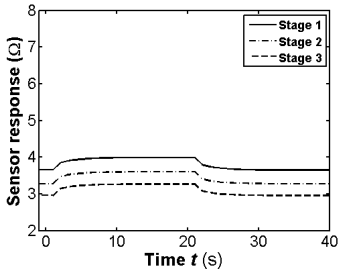
where $\mathrm{E}[\mathbf{r}(\mathbf{c}, t, s)]$ represents the expected responses from the deterministic simulation model (A.5), and $\boldsymbol{\varepsilon} = (\varepsilon_1, \varepsilon_2, \varepsilon_3)$ is the random error vector that follows multivariate normal distribution with mean zero and variance matrix $\Sigma_{\boldsymbol{\varepsilon}}$:



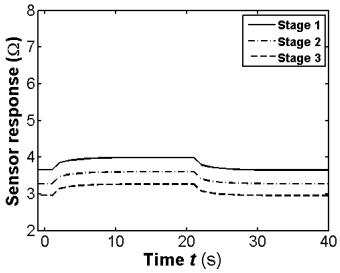
(a) Sensor 1 exposed to 300 ppm Analyte 1



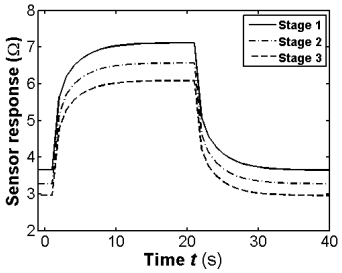
(b) Sensor 2 exposed to 300 ppm Analyte 1



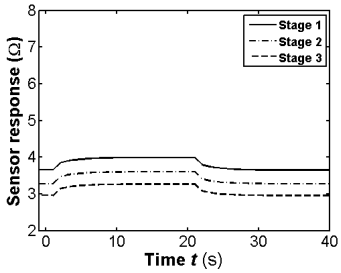
(c) Sensor 3 exposed to 300 ppm Analyte 1



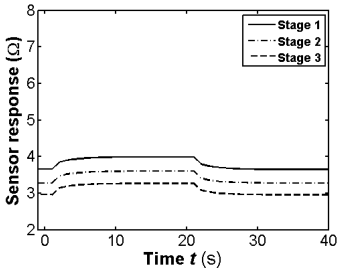
(d) Sensor 1 exposed to 300 ppm Analyte 2



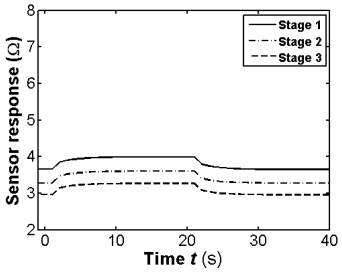
(e) Sensor 2 exposed to 300 ppm Analyte 2



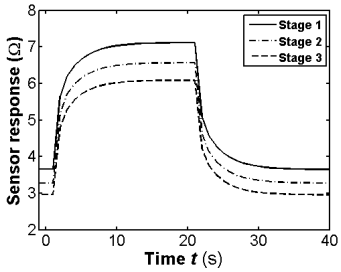
(f) Sensor 3 exposed to 300 ppm Analyte 2



(g) Sensor 1 exposed to 300 ppm Analyte 3



(h) Sensor 2 exposed to 300 ppm Analyte 3



(i) Sensor 3 exposed to 300 ppm Analyte 3

Figure 5.1 Expected response curves for the sensor array exposed to each analyte individually

$$\Sigma_{\boldsymbol{\varepsilon}} = \begin{bmatrix} 1 & 0.1 & 0.1 \\ 0.1 & 1 & 0.1 \\ 0.1 & 0.1 & 1 \end{bmatrix} \times 10^{-3} \tag{5.2}$$

5.2 Calibration of the Simulated Sensor Array

In this example, the sensor array is used to track the concentration changes of three target analytes in a sample mixture. The proportion of each analyte can be rapidly changing, as well as the total volume of the three. We assume the total concentration vary within the following range:

$$0 \text{ ppm} \leq c_1 + c_2 + c_3 \leq 300 \text{ ppm}. \tag{5.3}$$

There is assumed to be four usage stages in total, and the current one is Stage 4. For each stage, experimental data were collected following the design in Table 5.1, where the total volume of analytes is changing, as well as the proportion of individual analyte. Three replications are assigned to each of the distinct design points. For any replication, the experimental data are collected following the exposure cycle $t = 0, 1, 2, \ldots, 2T$ as in Figure 3.1. Specially, for different usage stages $s = 1, 2, \ldots$, we set a decreasing sample size in order to mimic the lack of experimental data in the drift detection process [9]. At current usage stage $s = 4$, the experiment effort is reaching its minimum, and the dynamic inverse model (4.14) can only be estimated based on the pseudo data provided by the fitted forward model (3.3).

For date pre-processing, it is a common practice to smooth out the array responses $\mathbf{r}(t)$ before applying the inverse model (4.14). Specifically, the simple moving average smoothing operator was used here for variability reduction:

$$\mathbf{r}_m(t) = \frac{\mathbf{r}(t-1) + \mathbf{r}(t) + \mathbf{r}(t+1)}{3}; \quad t = 1, 2, 3, \ldots, (2T-1). \tag{5.4}$$

Eventually, the available experimental data would be $(\mathbf{w}_i, \mathbf{r}_{mj}(\mathbf{w}_i))$ with $i = 1, 2, 3, \ldots, I$ and $j = 1, 2, \ldots, n$. Here, we have $I = D \times (s^* - 1) \times (2T - 1) = 12 \times 3 \times 39$ and $n = 3$. The total sample size is $I \times n$. Following the estimation methods in Figure 4.1, the fitted model parameters

are given as

$$\widehat{\Sigma}_{m\boldsymbol{\varepsilon}} = \begin{bmatrix} 3.32 & 0.30 & 0.32 \\ 0.30 & 3.28 & 0.32 \\ 0.32 & 0.32 & 3.30 \end{bmatrix} \times 10^{-4} \tag{5.5}$$

$$\widehat{\boldsymbol{\mu}} = \begin{bmatrix} 0.3583 \\ 0.3594 \\ 0.3572 \end{bmatrix}$$

$$\widehat{\Psi} = \begin{bmatrix} 1.6922 & 0.8459 & 0.8459 \\ 0.8459 & 1.6915 & 0.8457 \\ 0.8459 & 0.8457 & 1.6913 \end{bmatrix}$$

$$\widehat{\boldsymbol{\Theta}} = \begin{bmatrix} 0.0037 & 0.0001 & 0.0001 & 1.1508 & 0.0002 \\ 0.0001 & 0.0038 & 0.0001 & 1.1506 & 0.0002 \\ 0.0001 & 0.0001 & 0.0038 & 1.1506 & 0.0002 \end{bmatrix}$$

To generate the pseudo data for current stage (Step (3) Figure 3.2), we keep the same design in Table 5.1, and set $s = 4$. So, the total number of distinct experimental conditions $\mathbf{w}_0$ in the pseudo data is $D \times (2T-1)$. Related predictions are calculated by Function (4.13), and can be denoted as $\{(\widehat{\mathbf{c}}_i(t), \widehat{\mathbf{r}}_{mi}(t)); i = 1,2,\ldots,12; t = 1,2,\ldots,39\}$, from which we estimate the inverse dynamic model for real-time monitoring:

$$\begin{aligned} \widehat{c}_1(t) =& -215.85 + 97.41 r_{m1}(t) - 157.74 r_{m1}(t)^2 - 205.14 r_{m1}(t-1)^2 \\ &+ 363.48 r_{m1}(t) \cdot r_{m1}(t-1) - 9.42 r_{m2}(t) - 9.15 r_{m3}(t) \\ \widehat{c}_2(t) =& -222.96 - 9.57 r_{m1}(t) + 101.01 r_{m2}(t) - 147.39 r_{m2}(t)^2 \\ &- 192.21 r_{m2}(t-1)^2 + 339.78 r_{m2}(t) \cdot r_{m2}(t-1) - 8.91 r_{m3}(t) \\ \widehat{c}_3(t) =& -222.99 - 9.39 r_{m1}(t) - 9.36 r_{m2}(t) + 101.64 r_{m3}(t) - 163.2 r_{m3}(t)^2 \\ &- 210.6 r_{m3}(t-1)^2 + 373.83 r_{m3}(t) \cdot r_{m3}(t-1) \end{aligned} \tag{5.6}$$

Table 5.1 Design points of calibration experiments for the four usage stages

Stage 1			Stage 2			Stage 3			Stage 4		
c_1	c_2	c_3	c_1	c_2	c_3	c_1	c_2	c_3	c_1	c_2	c_3
60	0	0	60	0	0				60	0	0
0	60	0	0	60	0				0	60	0
0	0	60	0	0	60				0	0	60
30	30	0				30	30	0			
0	30	30				0	30	30			
30	0	30				30	0	30			
			40.2	9.9	9.9						
			9.9	40.2	9.9						
			9.9	9.9	40.2						
			20	20	20						
300	0	0	300	0	0						
0	300	0	0	300	0						
0	0	300	0	0	300						
150	150	0									
0	150	150									
150	0	150									
						201	49.5	49.5			
						49.5	201	49.5			
						49.5	49.5	201			
						100	100	100			

5.3 Sampling-based Evaluation

To evaluate the quality of the fitted dynamic calibration model, additional experiments were carried out at the current usage stage s^*, and it contains data different than and independent of those we used for model fitting. Two different validation data sets (VDS) are prepared mimicking different patterns of environment changes.

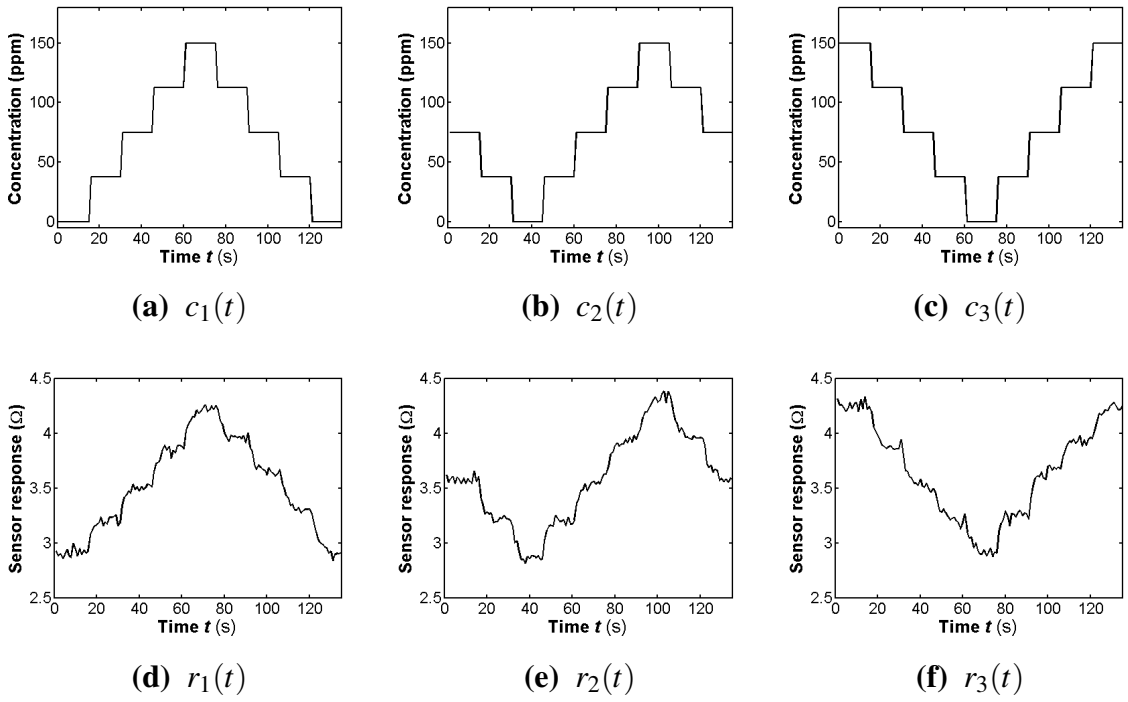
VDS I: Step-wisely Changing Concentration

The sensor array was exposed to the mixture of analytes whose concentrations follow Figure 5.2 (a-c) with the discrete time index $t = 0, 1, 2, \ldots, 135$ seconds. The observed array responses in the VDS are given in Figure 5.2 (d-f). The time-varying concentrations in Figure 5.2 (a-c) are the "true" concentrations that the sensor array has been exposed to.

Recall that $\mathbf{r}_m(t)$ is the smoothed sensor response vector (5.4). To evaluate the estimation ability of the fitted model (5.6), we first applied the moving average smoothing operator (5.4) on the original sensor responses $\{\mathbf{r}(t); t = 0, 1, \ldots, 135\}$ in the VDS. (Note that such smoothing can be easily done in the use of a sensor for real-time monitoring.) Then, the smoothed responses $\{\mathbf{r}_m(t); t = 1, 2, \ldots, 134\}$ derived from the VDS were fed to (5.6) to infer the underlying analytes' concentration $\{\widehat{\mathbf{c}}(t); t = 0, 1, \ldots, 133\}$. Through the recursive computation using $\{\mathbf{r}_m(t); t = 1, 2, \ldots, 134\}$ in the VDS, the $\{\widehat{\mathbf{c}}(t); t = 0, 1, \ldots, 133\}$ was obtained and plotted as the dashed curve in Figure 5.5. The solid curve in Figure 5.5 represents the "true" concentration, which is also depicted in Figure 5.2 (a-c). The shaded area in Figure 5.5 shows the 95% interval estimates (4.19) obtained by Algorithm GP-Bootstrapping (Figure 4.2). It suggests that, the probability of covering the unknown "true" concentration by this interval is 95%.

It is worthy of noting that sensor responses are subject to random variability, which is reflected in the zig-zag curve in Figure 5.2(d-f). When the sensor signal is fed to the inverse model (5.6) for concentration estimation, its variability is transformed through model (5.6) and carries over to the estimated concentration $\{\widehat{\mathbf{c}}(t); t = 0, 1, \ldots, 133\}$. The dashed curve (estimated analyte concen-

trations) in Figure 5.5 shows a similar zig-zag pattern, which is consistent with that of the sensor signals in Figure 5.2(d-f).



(a) $c_1(t)$ **(b)** $c_2(t)$ **(c)** $c_3(t)$

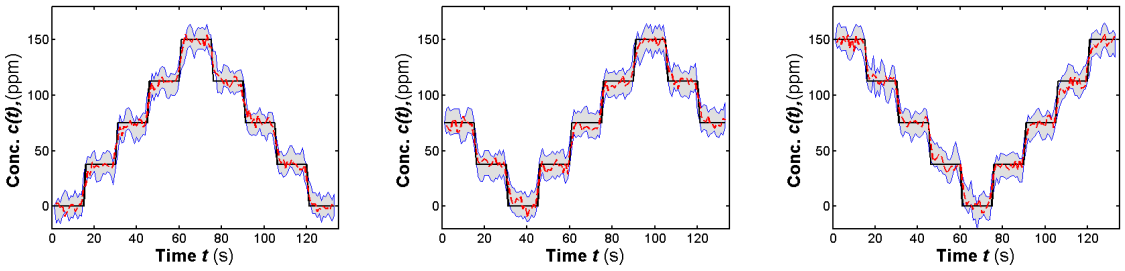
(d) $r_1(t)$ **(e)** $r_2(t)$ **(f)** $r_3(t)$

Figure 5.2 Experimental data in the validation data set (VDS I)

Evidently, when following our proposed calibration procedure, the sensor is able to provide a good concentration estimate in two aspects: (i) the quantification deviation caused by sensor drift has been successfully reduced, even with a shortage in experiment efforts; (ii) the dynamic calibration model (5.6) allows the sensor array to real-time track the changes in the analyte concentration. This is in contrast to the time of order of minutes that it takes for the sensor to reach a steady state.

VDS II: Continuously Changing Concentration

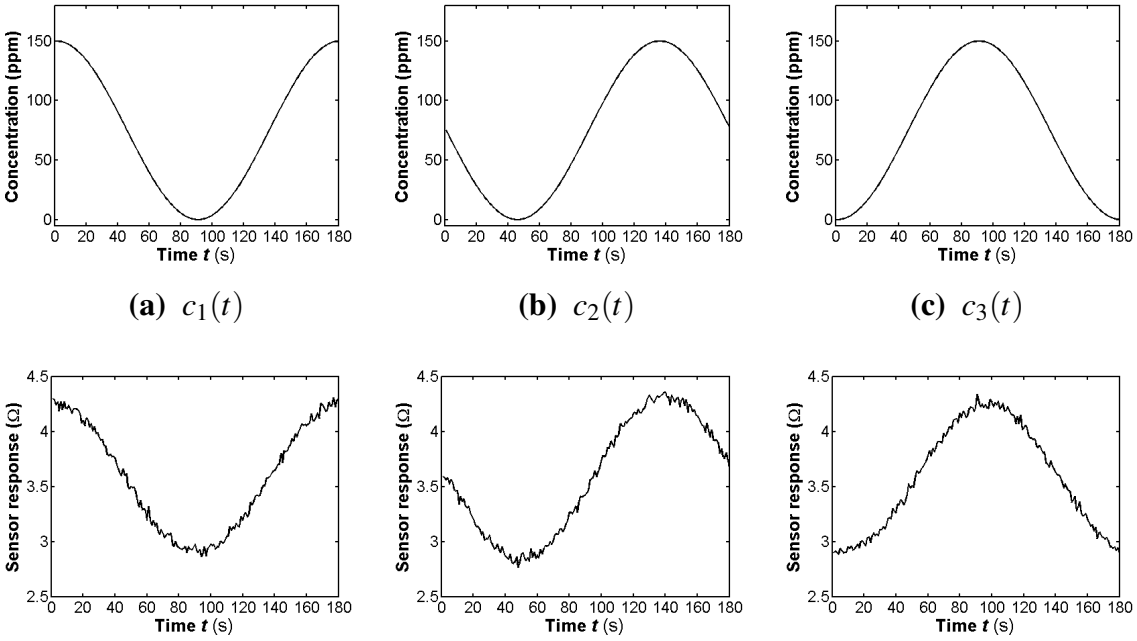
In a real environment, the analyte concentration may well vary continuously with time. To demonstrate the potential of our inverse model to track the analyte concentration that changes continuously with time, the validation data was generated as follows: The $\mathbf{c}(t)$ is specified as a sine function of time as shown in Figure 5.4(a-c), and the sensor array responses in Figure 5.4(d-



(a) Analyte 1 **(b)** Analyte 2 **(c)** Analyte 3

Figure 5.3 Comparison of the estimated concentrations and their true values in the validation data set (VDS I).

f).



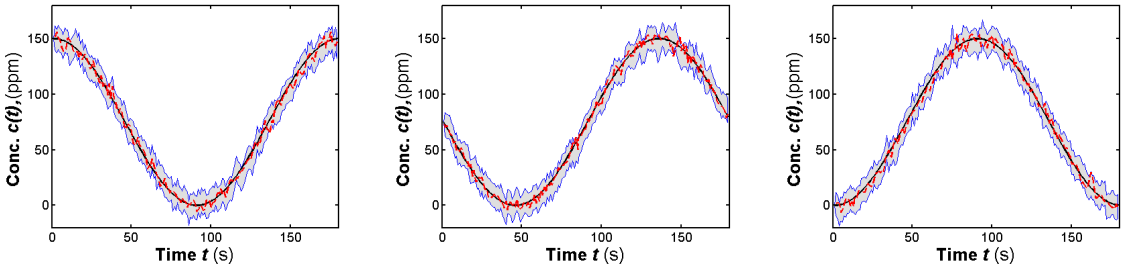
(a) $c_1(t)$ **(b)** $c_2(t)$ **(c)** $c_3(t)$

(d) $r_1(t)$ **(e)** $r_2(t)$ **(f)** $r_3(t)$

Figure 5.4 Experimental data in the validation data set (VDS II)

Let $\{\mathbf{r}_m(t); t = 1,2,\ldots,179\}$ be the smoothed sensor responses obtained from applying the moving average smoothing operator (5.4) on $\{\mathbf{r}(t); t = 0,1,\ldots,180\}$. $\{\mathbf{r}_m(t); t = 1,2,\ldots,179\}$

was fed to the inverse model (5.6) for recursive computation. The estimated analyte concentration $\{\widehat{\mathbf{c}}(t); t = 1, 2, \ldots, 178\}$ was calculated and plotted in Figure 5.5 as the dash line. The shaded area is the 95% interval estimate of the inferred concentration. The solid curve in Figure 5.5 represents the specified $\mathbf{c}(t)$ in VDS II which is also plotted in Figure 5.4(a-c).



(a) Analyte 1 **(b)** Analyte 2 **(c)** Analyte 3

Figure 5.5 Comparison of the estimated concentrations and their true values in the validation data set (VDS II).

As discussed in Section 4.2.2, the variability in simulated sensor responses (Figure 5.4 (d-f)) is transformed and passed on to the estimated analyte concentrations, which is reflected by the local zig-zag fluctuations in the dash line of Figure 5.5. Nevertheless, in Figure 5.5, the dash line are able to track the overall trend of the solid curve (i.e., the true time-varying concentration) with good accuracy throughout the entire period.

Chapter 6

Conclusions

The sensor array calibration model is a mathematical model relating the array response to the target analytes' concentrations. When coupled with the sensory device, the calibration model plays a critical role in quantifying the target analytes in an unknown environment, and the quality of the model directly affects the accuracy of the estimated analyte concentrations. Complex behaviors of chemical sensors, such as time-dependent outputs and drifting issues, have not been adequately addressed in the existing literature of sensor calibration. How to calibrate a drifting sensor array? How to deploy it for real-time monitoring of rapidly-changing environments? Statistical methods were developed in this work for both forward descriptive calibration and inverse dynamic calibration of sensor arrays.

The forward calibration model is fitted from experimental data, and assumes the form of a multivariate Gaussian process (GP). The multivariate GP synergistically models all calibration data collected under a range of drifting conditions, and seeks to produce the forward model of highest quality with the given experimental data. The multivariate GP also allows for valid statistical inference regarding the fitted model. Since the forward model is a descriptive model relating sensors' time-dependent responses to a static environment specified by several variables, it is not able to assist in real-time monitoring of rapidly-changing environments, which is why inverse dynamic

calibration method is also developed. The inverse model takes the form of a transfer function regression, infers the time-varying analyte concentrations from the dynamic sensor responses, and thus can be coupled with sensors for real-time monitoring. The inverse transfer function model is estimated from the pseudo calibration data generated by the forward multivariate GP model, which captures the sensors' dynamic and drifting behaviors as reflected in the real experimental data. A multivariate GP-based bootstrapping algorithm was adapted to quantify the uncertainty of the analyte concentrations estimated by the inverse model. The calibration method were applied to calibrate a simulated sensor array with drifting behaviors, and its effectiveness has been demonstrated.

This work focuses on the forward and inverse modeling with given experimental calibration data. A natural future step is to develop efficient design of experiments (DOE) method based on the forward GP and inverse dynamic modeling. The DOE will seek to minimize the experimental effort needed at the current usage stage of the sensor array to ensure the desired quality of the calibration models.

Appendix A

Simulation Models for the Drifting Sensor Array

In this appendix, we firstly discuss some basic knowledges about the dynamic performance of chemiresistors; then a complete simulation algorithm will be given.

A.1 Characteristics of Chemiresistors

When exposed to analytes, most chemiresistors will respond rapidly at first and then slowly reach their steady-state [66]. For a single sensor, the general form of the response function can be written as [28]

$$\mathrm{E}[r(t)] = b + a \cdot c \cdot g(t) \tag{A.1}$$

where b is the baseline signal and a the parameter of sensitivity. Both b and a would drift over long terms of use [28]. Here, $g(t)$ is the dynamic decay function only related to the sampling time t. Specifically, we adopt an exponential decay function with two significantly different exponential components, because it can effectively simulate both steady-state and dynamic characteristics of sensors [10]. When the concentration jumps from 0 to c after the sampling time $t = 0$, the sensor

response shows an increasing trend as

$$g(t) = g^{inc}(t) = [1 - \alpha_1 \exp(-\frac{t}{\tau_1}) - \alpha_2 \exp(-\frac{t}{\tau_2})] \tag{A.2}$$

with α and τ being positive constants determined by the given sensing material. It always holds that $\alpha_1 + \alpha_2 = 1$. It is clear that the steady-state response can be reached when $g^{inc}(t) = 1$, and the exact value is

$$\mathrm{E}[r] = b + a \cdot c. \tag{A.3}$$

On the other hand, if the concentration drops from c to 0 at $t > 0$, the dynamic function $g(t)$ will give a decrease form as:

$$g(t) = g^{dec}(t) = [\alpha_1 \exp(-\frac{t}{\tau_1}) + \alpha_2 \exp(-\frac{t}{\tau_2})] \tag{A.4}$$

with the same constants α and τ as in (A.2).

A.2 Simulated Sensor Array

Our simulated sensor array consists of three sensors while targeting on three analytes. The general form is

$$\begin{aligned}
\mathrm{E}[r_1(t)] =& b_1 + a_{11} \cdot c_1 \cdot g_{11}(t) + a_{21} \cdot c_1 \cdot g_{21}(t) + a_{31} \cdot c_1 \cdot g_{31}(t) \\
\mathrm{E}[r_2(t)] =& b_2 + a_{12} \cdot c_1 \cdot g_{12}(t) + a_{22} \cdot c_1 \cdot g_{22}(t) + a_{32} \cdot c_1 \cdot g_{32}(t) \\
\mathrm{E}[r_3(t)] =& b_3 + a_{13} \cdot c_1 \cdot g_{13}(t) + a_{23} \cdot c_1 \cdot g_{23}(t) + a_{33} \cdot c_1 \cdot g_{33}(t)
\end{aligned} \tag{A.5}$$

where $g_{pq}(t)$ is the dynamic decay function for Sensor q on Analyte p, and a_{pq} the cross-sensitivity. Drifting effects are represented by the different values of b, a and τ. See below for the complete model, whose parameters are based on the three years' continuous measurement on drifting sensor array. This experimental data set is the most comprehensive in drifting sensor study, and provided by Vergara et al. [27].

$$
\begin{aligned}
\mathrm{E}[r_1(t)] =& 0.98 + 3.10\exp(-0.15s) \\
&+ [1.77 + 1.90\exp(-0.11s)]\, c_1 g_{11}(t) \\
&+ [0.18 + 0.19\exp(-0.11s)]\, c_2 g_{21}(t) \\
&+ [0.18 + 0.19\exp(-0.11s)]\, c_3 g_{31}(t) \\
\mathrm{E}[r_2(t)] =& 0.98 + 3.10\exp(-0.15s) \\
&+ [0.18 + 0.19\exp(-0.11s)]\, c_1 g_{12}(t) \\
&+ [1.77 + 1.90\exp(-0.11s)]\, c_2 g_{22}(t) \\
&+ [0.18 + 0.19\exp(-0.11s)]\, c_3 g_{32}(t) \\
\mathrm{E}[r_3(t)] =& 0.98 + 3.10\exp(-0.15s) \\
&+ [0.18 + 0.19\exp(-0.11s)]\, c_1 g_{13}(t) \\
&+ [0.18 + 0.19\exp(-0.11s)]\, c_2 g_{23}(t) \\
&+ [1.77 + 1.90\exp(-0.11s)]\, c_3 g_{33}(t)
\end{aligned}
\tag{A.6}
$$

For increasing responses, we have

$$
g_{pq}^{inc}(t) = \left[1 - 0.5\exp\left(\frac{-t}{2.18 + 0.91\exp(-0.1s)}\right) - 0.5\exp\left(\frac{-t}{0.36 + 0.15\exp(-0.1s)}\right)\right], \tag{A.7}
$$

and the decreasing one is

$$
g_{pq}^{dec}(t) = \left[0.5\exp\left(\frac{-t}{2.18 + 0.91\exp(-0.1s)}\right) + 0.5\exp\left(\frac{-t}{0.36 + 0.15\exp(-0.1s)}\right)\right] \tag{A.8}
$$

To cope with the continuously changing concentrations $\mathbf{c}(t)$, a recursive algorithm is developed in Figure A.1 and will be used throughout our simulation study.

Input: (i) The sensor's current usage stage s^*;

(ii) The analyte' concentration $\mathbf{c}(t); t = 0, 1, 2, \ldots, T$. The sensors hold their steady-state at $t = 0$

Initialize Get all parameters updated with $s = s^*$;

Set an intermediate variable $u_{pq}(0) = s_{pq} \cdot c_p(t = 0)$, for $p = 1, 2, 3; q = 1, 2, 3$

Do for $t = 1, 2, \ldots, T$

(1) IF $u_{pq}(t-1) \leq s_{pq} \cdot c_p(t)$, then select $g_{pq}(t) = g_{pq}^{inc}(t)$

ELSE select $g_{pq}(t) = g_{pq}^{dec}(t)$

(2) Use numerical search method to find t'_{pq}, which satisfies $s_{pq} \cdot c_p(t) \cdot g(t'_{pq}) = u_{pq}(t-1)$.

(3) Get $u_{pq}(t) = s_{pq} \cdot c_p(t) \cdot g_{pq}(t' + 1)$

(3) Return $\mathrm{E}[r_q(t)] = b_q + \sum_p^3 s_{pq} \cdot c_p(t) \cdot g_{pq}(t'_{pq} + 1) + \varepsilon_q; q = 1, 2, 3$

CONTINUE

Output: $\mathrm{E}[\mathbf{r}(t)]; t = 1, 2, 3, \ldots, T$

Figure A.1 Simulate the time-dependent responses with time-varying concentrations

Appendix B

Nomenclature

$\mathbf{c}$	the variable vector representing the analyte concentrations of an environment
c_p	the concentration of the p^{th} component analyte
P	the number of target analytes
$\mathbf{c}_0$	the true concentration vector for the analytes in an environment
$\mathbf{r}$	the random vector representing the sensor array responses
r_q	the response of the q^{th} component in the array
Q	the number of sensors in the array
t	the exposure time index
s	the usage stage, reflecting the level of sensor drift
$\mathbf{w}$	the experimental conditions in sampling
H	the dimension of the vector $\mathbf{w}$
$\boldsymbol{\varepsilon}$	the random error vector for array responses

$\mathbf{F}(\mathbf{w})$	the forward calibration model based on multivariate Gaussian Processes
$\mathbf{G}(\cdot)$	the inverse dynamic model taking the form of transfer functions
$\Sigma_{\boldsymbol{\varepsilon}}$	the variance matrix of the random error vector $\boldsymbol{\varepsilon}$
$\mathsf{M}(\cdot)$	stationary multivariate Gaussian Processes
$\boldsymbol{\mu}$	the mean vector of the multivariate Gaussian Processes
Ψ	the constant variance matrix of the multivariate Gaussian Processes
$\boldsymbol{\Theta}$	the correlation parameters of the multivariate Gaussian Processes
$\mathbf{e}$	the residual vector of the forward calibration model
B	the time order of the inverse dynamic model $\mathbf{G}(\cdot)$
N_B	the bootstrapping resampling sizes
D	the number of distinct design points on $\mathbf{c}$
I	the number of distinct experimental conditions $\mathbf{w}$
n	the number of replications
$\mathbf{r}_m(t)$	the smoothed array response obtained by moving-average

Bibliography

[1] C. E. Rasmussen and C. K. I. Williams. *Gaussian Processes for Machine Learning*. the MIT Press, Boston, 2006.

[2] J. R. Stetter and W. Penrose. *Understanding Chemical Sensors and Chemical Sensor Arrays (Electronic Noses): Past, Present, and Future*, volume 10, page 189. Wiley-VCH: Weinheim, Germany, third edition, 2002.

[3] N. Docquier and S. Candel. Combustion control and sensors: a review. *Progress in Energy and Combustion Science*, 28:107–150, 2002.

[4] U. Guth, W. Vonau, and J. Zosel. Recent developments in electrochemical sensor application and technology-a review. *Meas. Sci. Technol.*, 20:042002, 2009.

[5] N. Wu, M. Zhao, J. G. Zheng, C. Jiang, B. Myers, M. Chyu, S. Li, and S. X. Mao. Porous CuO-ZnO nanocomposite for sensing electrode of high-temperature CO solid-state electrochemical sensor. *Nanotechnology*, 16:2878–2881, 2005.

[6] M. Holmberg, F. Winquist, I. Lundström, F. Davide, C. DiNatale, and A. D'Amico. Drift counteraction for an electronic nose. *Sensors and Actuators B: Chemical*, 36(1-3):528–535, 1996.

[7] J. E. Haugen, O. Tomic, and K. Kvaal. A calibration method for handling the temporal drift of solid state gas-sensors. *Analytica Chimica Acta*, 407:23–39, 2000.

[8] M. Holmberg and T. Artursson. Drift compensation, standards, and calibration methods. In *Handbook of Machine Olfaction: Electronic Nose Technology*, pages 325–346. Wiley-VCH, 2003.

[9] S. Marco and A. Gutiérrez-Gálvez. Signal and data processing for machine olfaction and chemical sensing: A review. *IEEE Sensors Journal*, 12:3189–3213, 2012.

[10] J. Samitier, J. M. López-Villegas, S. Marco, L. Cámara, A. Pardo, O. Ruiz, and J. R. Morante. A new method to analyze signal transients in chemical sensors. *Sensors and Actuators B*, 18(1-3):308–312, 1994.

[11] D. M. Wilson and S. P. Deweerth. Odor discrimination using steady-state and transient characteristics of tin-oxide sensors. *Sensors and Actuators B*, 28(2):123–128, 1995.

[12] X. Vilanova, E. Llobet, R. Alcubilla, J. E. Sueiras, and X. Correig. Analysis of the conductance transient in thick-film tin oxide gas sensors. *Sensors and Actuators B: Chemical*, 31(3):175–180, 1996.

[13] E. Llobet, J. Brezmes, X. Vilanova, J. E. Sueiras, and X. Correig. Qualitative and quantitative analysis of volatile organic compounds using transient and steady-state responses of a thick-film tin oxide gas sensor array. *Sensors and Actuators B*, 41(1-3):13–21, 1997.

[14] R. Gutierrez-Osuna, H. T. Nagle, and S. S. Schiffman. Analysis of the conductance transient in thick-film tin oxide gas sensors. *Sensors and Actuators B*, 61(1-3):170–182, 1999.

[15] L. Carmel, S. Levy, D. Lancet, and D. Harel. A feature extraction method for chemical sensors in electronic noses. *Sensors and Actuators B*, 93:67–76, 2003.

[16] E. Martinelli, C. Falconi, A. D'Amico, and C. Di Natale. Feature extraction of chemical sensors in phase space. *Sensors and Actuators B*, 95:132–139, 2003.

[17] M. K. Muezzinoglu, A. Vergara, R. Huerta, N. Rulkov, M. I. Rabinovich, A. Selverston, and H. D. Abarbanel. Acceleration of chemo-sensory information processing using transient features. *Sensors and Actuators B*, 137:507–512, 2009.

[18] A. C. Romain and J. Nicolas. Long term stability of metal oxide-based gas sensors for e-nose environmental applications: An overview. *Sensors and Actuators B*, 246:502–506, 2010.

[19] F. Hossein-Babaei and V. Ghafarinia. Compensation for the drift-like terms caused by environmental fluctuations in the responses of chemoresistive gas sensors. *Sensors and Actuators B*, 143:641–648, 2010.

[20] M. Ghasemi-Varnamkhasti, S. S. Mohtasebi, M. S., J. Lozano, H. Ahmadi, S. H. Razavi, and A. Dicko. Aging fingerprint characterization of beer using electronic nose. *Sensors and Actuators B: Chemical*, 159(1):51–59, 2011.

[21] M. L. Frank, M. D. Fulkerson, B. R. Patton, and P. K. Dutta. TiO_2-based sensor arrays modeled with nonlinear regression analysis for simultaneously determining CO and O_2 concentrations at high temperatures. *Sensors and Actuators B: Chemical*, 87:471–479, 2002.

[22] P. Zhang, C. Lee, H. Verweij, SA. Akbar, G. Hunter, and PK. Dutta. High temperature sensor array for simultaneous determination of O_2, CO, and CO_2 with kernel ridge regression data analysis. *Sensors and Actuators B*, 123:950–963, 2007.

[23] B. Curry and P. H. Morgan. Model selection in neural networks: some difficulties. *European Journal of Operations Research*, 170(2):567–577, 2006.

[24] G. P. Zhang. Avoiding pitfalls in neural network research. *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews*, 37(1):3–16, 2007.

[25] O. Svensson, T. Kourti, and J. F. MacGregor. An investigation of orthogonal signal correction algorithms and their characteristics. *Journal of Chemometrics*, 16(4):176–188, 2002.

[26] P. Gujral, M. Amrhein, and D. Bonvin. Drift correction in multivariate calibration models using on-line reference measurements. *Analytica Chimica Acta*, 642(1-2):27–32, 2009.

[27] A. Vergara, S. Vembu, T. Ayhan, M. A. Ryan, M. L. Homer, and R. Huerta. Chemical gas sensor drift compensation using classifier ensembles. *Sensors and Actuators B: Chemical*, 166:320–329, 2012.

[28] M. J. Wenzel, A. Mensah-Brown, F. Josse, and E. E. Yaz. Online drift compensation for chemical sensors using estimation theory. *IEEE Sensors Journal*, 11:225–232, 2011.

[29] N. Cressie. Kridging non-stationary data. *Journal of American Statistics Association*, 81:625–634, 1986.

[30] N. Cressie and C. K. Wikle. *Statistics for Spatio-Temporal Data*. Wiley, Hoboken, N. J., 2011.

[31] J. Sacks, W. J. Welch, T. J. Mitchell, and H. P. Wynn. Design and analysis of computer experiments. *Statistical Science*, 4(4):409–423, 1989.

[32] G. M. Laslett. Kriging and splines: An empirical comparison of their predictive performance in some applications. *Journal of the American Statistical Association*, 89:391–400, 1994.

[33] J. Oakley. *Bayesian Uncertainty Analysis For Complex Computer Codes*. PhD thesis, University of Sheffield, 1999.

[34] B. Ankenman, B. L. Nelson, and J. Staum. Stochastic kriging for simulation metamodeling. *Operations Research*, 58(2):371–382, 2010.

[35] J. Lehman. *Sequential Design of Computer Experiments for Robust Parametric Design*. PhD thesis, Department of Statistics, Ohio State University, Columbus, OH, 2002.

[36] T. J. Santner, B. J. Williams, and W. I. Notz. *The Design and Analysis of Computer Experiments*. Springer, New York, 2003.

[37] J. Loeppky, L. Moore, and B. Williams. Batch sequential designs for computer experiments. *Journals of Statistical Planning and Inference*, 140:1452–1464, 2010.

[38] Z. Geng, F. Yang, X. Chen, and N. Wu. Gaussian process based modeling and experimental design for sensor calibration in drifting environments. *Sensors and Actuators B: Chemical*, 2015.

[39] Z. Zhang. *New modeling procedures for functional data in computer experiments*. PhD thesis, The Pennsylvania State University, 7 2007.

[40] S. Conti and A. O'Hagan. Bayesian emulation of complex multi-output and dynamic computer models. *Journal of Statistical Planning and Inference*, 140:640–651, 2010.

[41] P. Boyle and M. Frean. Dependent gaussian processes. In *Advances in Neural Information Processing Systems 17*, pages 217–224. MIT Press, 2005.

[42] A. Majumdar and A. E. Gelfand. Multivariate spatial modeling for geostatistical data using convolved covariance functions. *Mathematical Geology*, 39(2):225–245, 2007.

[43] R. Hankin. Introducing multivator: A multivariate emulator. *Journal of Statistical Software*, 46(8), 2012.

[44] T. E. Fricker, J. E. Oakley, and N. M. Urban. Multivariate gaussian process emulators with nonseparable covariance structures. *Technometrics*, 55:47–âĂŞ56, 2013.

[45] G. E. P. Box and G. M. Jenkins. *Time Series Analysis: Forecasting and Control*. Holden-Day, Oakland., 1976.

[46] P. J. Brown. Multivariate calibration. *Journal of the Royal Statistical Society B*, 44:287–321, 1982.

[47] R. Sundberg. Multivariate calibration - direct and indirect regression methodology. *Scandinavian Journal of Statistics*, 26:161–207, 1999.

[48] H. Lei, W. G. Pitt, L. K. McGrath, and C. K. Hob. Modeling carbon black/polymer composite sensors. *Sensors and Actuators B*, 125:396–407, 2007.

[49] Z. Geng, F. Yang, and M. Li. A bootstrapping-based statistical procedure for multivariate calibration of sensor arrays. *Sensors and Actuators B: Chemical*, 188:440–453, 2013.

[50] D. E. Goldberg. *Genetic Algorithms in Search, Optimization, and Machine Learning*. Addison-Wesley Professional, 1989.

[51] F. R. Saphiro, SD. Senturia, and D. Adler. The use of linear predictive modeling for the analysis of transients from experiments on semiconductor defects. *Journal of Applied Physics*, 55(10):3453–3459, 1984.

[52] H. Barkhuijsen, R. Debeer, W. M. M. J. Bovee, and D. Vanormondt. Retrieval of frequencies, amplitudes, damping factors, and phases from time-domain signals using a linear least-squares procedure. *Journal of Magnetic Resonance*, 61(3):465–481, 1985.

[53] A. V. Oppenheim and R. W. Schafer. *Discrete-Time Signal Processing*. Prentice Hall, 3 edition, 2009.

[54] M. H. Kutner, C. J. Nachtsheim, and J. Neter. *Applied Linear Regression Models*. McGraw-Hill/Irwin, 4 edition, 2004.

[55] N. Wu, Z. Chen, J. Xu, M. Chyu, and S. X. Mao. Impedance-metric Pt/YSZ/Au-Ga_2O_3 sensor for CO detection at high-temperature. *Sensors and Actuators B-Chemical*, 110:49–53, 2005.

[56] G. Jones. Bootstrapping in controlled calibration experiments. *Technometrics*, 41:224–233, 1999.

[57] B. Efron and R. J. Tibshirani. *An Introduction to the Bootstrap*. Chapman and Hall, New York, 1993.

[58] D. Malzahn and M. Opper. Learning curves and bootstrap estimates for inference with gaussian processes: A statistical mechanics study. *Complexity*, 8:57–63, 2003.

[59] D. den Hertog and J. Kocijan. Customized sequential designs for random simulation experiments: Kriging metamodeling and bootstrapping. *European Journal of Operational Research*, 186(3):1099–1113, 2008.

[60] Wim C.M. van Beers and Jack P.C. Kleijnen. Customized sequential designs for random simulation experiments: Kriging metamodeling and bootstrapping. *European Journal of Operational Research*, 186(3):1099–1113, 2008.

[61] P. D. W. Kirk and M. P. H. Stumpf. Gaussian process regression bootstrapping: exploring the effects of uncertainty in time course data. *Bioinformatics*, 25(10):1300–1306, 2009.

[62] MS. Srivastava. *Methods of Multivariate Statistics*. Wiley-Interscience, 2002.

[63] A. Law. *Simulation Modeling and Analysis with Expertfit Software*. McGraw-Hill series in industrial engineering and management science. McGraw-Hill Education, fourth edition, 2006.

[64] K. Varmuza and P. Filzmoser. *Introduction to multivariate statistical analysis in chemometrics*. CRC Press,Boca Raton, FL, 2009.

[65] Z. Geng, F. Yang, and N. Wu. Optimum design of sensor arrays via simulation-based multivariate calibration. *Sensors and Actuators B: Chemical*, 156:854–862, 2011.

[66] F. Yang, Z. Geng, A. Koneru, M. Zhi, H. Li, and N. Wu. Dynamic calibration of electrochemical sensor for accelerated analyte quantification. *IEEE Sensors Journal*, 13:1192–1199, 2013.