A DYNAMIC BAYESIAN NONPARAMETRIC MODEL FOR BLIND CALIBRATION OF SENSOR NETWORKS

Jielong Yang[⋆] *Wee Peng Tay*[⋆] *Xionghu Zhong*[⋆]

[⋆] School of Electrical and Electronic Engineering
Nanyang Technological University,Singapore

ABSTRACT

In the sensor network blind calibration problem, the gains and offsets of sensors are estimated from noisy observations of unknown underlying signals. This is in general a non-identifiable problem, unless restrictive assumptions on the signal subspace or sensor observations are imposed. To overcome these assumptions, we propose a dynamic Bayesian nonparametric model. We show that if the unknown underlying signals follow the first-order auto-regressive process, then the sensor gains and offsets are identifiable. Furthermore, our model allows sensors to form clusters, where each cluster observes the same underlying signal. The clusters are however not known a priori, and are learned through the sensor data. We present a block Gibbs sampling inference method based on the forward filtering backward sampling algorithm. Simulation results suggest that our approach can estimate the sensor gains and offsets with good accuracy, and performs better than methods that first perform clustering and then blind calibration.

***Index Terms*—** Blind calibration, dynamic Bayesian nonparametrics, Dirichlet process, block Gibbs sampling

1. INTRODUCTION

Because of differences in the materials and electrical components that make up a sensor and other environmental factors, each sensor has a gain and offset that needs to be estimated or calibrated before its observations can be interpreted meaningfully [1,2]. For example, in using sensors to perform monitoring of a physical phenomenon, modeling of the distributions governing the sensor observations is required in various distributed inference methods [3,4]. A sensor can be calibrated in a controlled environment by utilizing its observations of a known signal to estimate its gain and offset. However, this is a painstaking task, which may be impractical if the sensor network is large, and sensors may be placed in inaccessible locations. Macro calibration has thus been proposed to calibrate an entire sensor network based on observations from all sensors in the network [5, 6]. Furthermore, the calibration is done without knowledge of the underlying signals that the sensors are observing. This is known as *blind* calibration [1,7]. Since without knowing the ground truth signals, the sensor gains and offsets are non-identifiable if estimation is to be done solely based on the sensors' noisy observations. As such, additional assumptions are required.

In [1], a blind calibration method is proposed based on the assumption that the underlying signals observed by all sensors are deterministic and lie in a known subspace, and the time-averaged measurements of every sensor approaches a known mean value. The paper [8] proposes a distributed blind calibration algorithm and assumes that all the sensors observe the same stochastic underlying signal. In some applications in which sensors observe multiple signals, the aforementioned approach cannot be applied if we do not know a priori which sensor is observing which signal. In [9, 10], blind calibration methods are based on redundancy information provided by co-located sensors, which are classified into a group and assumed to have correlated measurements. Both of these two methods assume that sensors are densely deployed.

In this paper, we adopt a Bayesian nonparametric approach to sensor blind calibration. We assume that sensors are monitoring an unknown number of signals, and the signal observed by each sensor is unknown. We say that two sensors are in the same cluster if they observe the same signal, which follows a first-order auto-regressive (AR) stochastic process. This assumption is reasonable in many applications, including multi-region temperature monitoring and multi-object tracking. We show that with this assumption, the sensor gains and offsets are identifiable. We then propose a dependent dynamic model inspired by the Dirichlet process for the sensor observations: At each time t, the sensor observations follow a Bayesian nonparametric model, and the model at time t depends on the model at time $t-1$. To the best of our knowledge, the use of a dependent Bayesian nonparametric model for sensor network blind calibration is novel. We present block Gibbs sampling inference method, and perform simulations to compare the performance of our approach with approaches that first perform clustering and then blind calibration. Unlike [1, 8–10], we do not require sensors to be densely deployed or to know the time-averaged measurements of each sensor.

The rest of this paper is organized as follows. In Section 2, we present our problem formulation and sufficient conditions under which the sensor gains and offsets are identifiable. In Section 3, we propose a dynamic Bayesian nonparametric model for blind calibration. In Section 4, we develop inference methods for our model. Simulation results are presented in Section 5, and we conclude in Section 6. In the rest of this paper, we use $\mathcal{N}(\mu, \sigma^2)$ to represent the Gaussian distribution with mean μ and variance σ^2. The notation $y \mid x$ denotes a random variable y conditioned on x, and $p(y \mid x)$ denotes its conditional probability density function. We use $\sim$ to represent equality in distribution. The sequence $y(1), y(2), \ldots, y(t)$ is written as $y(1:t)$.

2. PROBLEM FORMULATION AND IDENTIFIABILITY

In this section, we present the blind calibration problem, our system model and assumptions, and show that the sensor gains and offsets are identifiable under our assumptions.

Consider N sensors monitoring an unknown number of signals $\{\theta_k(t) : k \geq 1\}$ over a period of time $t = 1, 2, \ldots, T$. The signal observed by each sensor n is also unknown. At each time t, sensor

n observes

$$y_n(t) = \alpha_n \theta_{s_n(t)}(t) + \beta_n + w_n(t), \tag{1}$$

where α_n and β_n are the gain and offset of sensor n respectively, $s_n(t)$ is the index of the signal sensor n is monitoring, and $w_n(t)$ is an additive observation noise with distribution $\mathcal{N}(0, \varsigma_n^2(t))$. The observation noises $w_n(t)$ are independent across sensors and time. Estimating α_n and β_n without knowing $\theta_{s_n(t)}(t)$ a priori is known as blind calibration. This is in general a non-identifiable estimation problem since $\alpha_n \theta_{s_n(t)} = (c\alpha_n)(\theta_{s_n(t)}/c)$ for any constant $c \neq 0$. In the following, we provide a sufficient condition for identifiability of the sensor gain and offset.

Proposition 1. *For each $n \geq 1$, suppose that for some $t \geq 1$, we have $s_n(t') = s_n(t) = k$ for $t < t' \leq t+2$ and some k. Suppose that*

$$\theta_k(t') = \lambda_k \theta_k(t'-1) + \epsilon_k(t'), \tag{2}$$

where $\lambda_k \neq 0$ and $\epsilon_k(t')$ are independent random variables with zero mean and variance $\sigma^2 > 0$. Suppose also that (2) holds for $t' = 1$ and some $k = s_n(1)$, where $\theta_{s_n(1)}(0)$ has mean $m_{s_n(1)}(0)$. Then, the parameters (α_n, β_n) are identifiable.

Proof. Suppose that (α_n, β_n) is not identifiable. Let $(\alpha_n', \beta_n') \neq (\alpha_n, \beta_n)$ be another sensor gain and offset that give rise to the same joint distribution for the sensor observations $y_n(t)$ and signals $\theta_{s_n(t)}(t)$ for all t. Let $(m_k(t), v_k(t))$ and $(m_k'(t), v_k'(t))$ be the mean and variance of the linear least squares (LLS) estimator of $\theta_k(t) \mid y_n(1:t)$ when the sensor parameters are (α_n, β_n) and (α_n', β_n'), respectively. Then from Chapter 18 of [11], the LLS predictor of $y_n(t+1) \mid y_n(1:t)$ when the sensor parameters are (α_n, β_n) has variance

$$\alpha_n^2 \lambda_k^2 v(t) + \alpha_n^2 \sigma^2 + \varsigma_n^2(t+1), \tag{3}$$

and the variance of the LLS estimator of $\theta_k(t+1) \mid y_n(1:t+1)$ is

$$v_k(t+1) = \lambda_k^2 v_k(t) + \sigma^2 - \frac{\alpha_n^2(\lambda_k^2 v_k(t) + \sigma^2)^2}{\alpha_n^2 \lambda_k^2 v_k(t) + \alpha_n^2 \sigma^2 + \varsigma_n^2(t+1)}. \tag{4}$$

Similar expressions hold for (α_n', β_n'). From (3), since α_n and α_n' induces the same joint distribution and hence the same LLS estimator and predictor, we have

$$\alpha_n^2 \lambda_k^2 v_k(t) + \alpha_n^2 \sigma^2 = {\alpha_n'}^2 \lambda_k^2 v_k'(t) + {\alpha_n'}^2 \sigma^2, \tag{5}$$

which, together with (4) leads to

$$\begin{aligned} &\alpha_n^2 \lambda_k^2 v_k(t+1) \\ &= \lambda_k^2 \left(\alpha_n^2 \lambda_k^2 v_k(t) + \alpha_n^2 \sigma^2 - \frac{(\alpha_n^2 \lambda_k^2 v_k(t) + \alpha_n^2 \sigma^2)^2}{\alpha_n^2 \lambda_k^2 v_k(t) + \alpha_n^2 \sigma^2 + \varsigma_n^2(t+1)} \right) \\ &= {\alpha_n'}^2 \lambda_k^2 v_k'(t+1). \end{aligned} \tag{6}$$

Applying (6) to (5) with t replaced by $t+1$, we obtain $\alpha_n = \alpha_n'$, a contradiction. Therefore α_n is identifiable. If the sensor parameters are (α_n, β_n), the mean of $y_n(1)$ is $\alpha_n \lambda_k m_{s_n(1)}(0) + \beta_n$, whereas the mean is $\alpha_n' \lambda_k m_{s_n(1)}(0) + \beta_n'$ if the sensor parameters are (α_n', β_n'). Since $\alpha_n = \alpha_n'$, we obtain $\beta_n = \beta_n'$.

The proof is complete. □

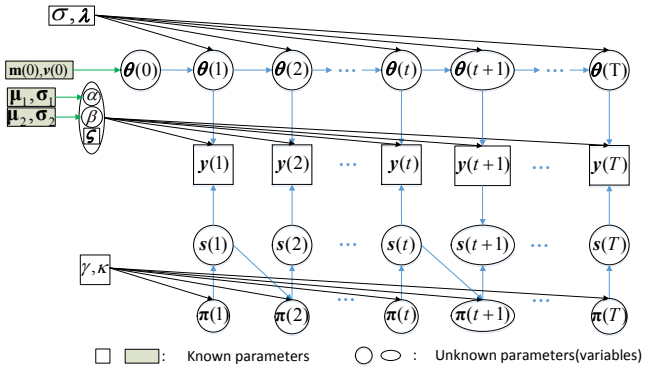

Fig. 1. Graphical representation of the dynamic Bayesian nonparametric model.

3. DYNAMIC BAYESIAN NONPARAMETRIC MODEL

In this section, we propose a dynamic Bayesian nonparametric model to couple the sensor signal evolution with its observations. Fig. 1 shows the graphical representation of our model. Let $\boldsymbol{\alpha} = (\alpha_n)_{n=1}^N$, $\boldsymbol{\beta} = (\beta_n)_{n=1}^N$, $\boldsymbol{\varsigma}(t) = (\varsigma_n(t))_{n=1}^N$, $\boldsymbol{\lambda} = \{\lambda_k : k \geq 1\}$, $\mathbf{y}(t) = (y_n(t))_{n=1}^N$, and $\boldsymbol{\theta}(t) = (\theta_k(t))_{k=1}^\infty$. The priors of α_n and β_n are

$$\alpha_n \sim \mathcal{N}\left(\mu_{1,n},\ \sigma_{1,n}^2\right),\ \beta_n \sim \mathcal{N}\left(\mu_{2,n},\ \sigma_{2,n}^2\right) \tag{7}$$

where $\mu_{1,n}, \sigma_{1,n}, \mu_{2,n}, \sigma_{2,n}$ are known hyperparameters. We note that Gaussian priors for the calibration parameters are widely adopted in the literature, see [9, 12, 13]. We assume that for each $k \geq 1$, $\theta_k(t)$ follows the first-order AR process given in (2), where we now also assume that $\epsilon_k(t) \sim \mathcal{N}\left(0,\ \sigma^2\right)$ for all $t \geq 1$. Let $\mathbf{m}(0)$ and $\mathbf{v}(0)$ be the mean and covariance of $\boldsymbol{\theta}(0)$, which is also assumed to be Gaussian distributed. For each node, we assume its gain, offset and the true signal it observes are mutually independent.

At each time t, sensors observing the same signal $\theta_k(t)$ can be considered to be clustered into the same group, where $s_n(t) = k$ indicates the index of the cluster that sensor n belongs to. Let $\pi_{n,k}(t)$ be the probability that $s_n(t) = k$, i.e., $s_n(t) \mid \boldsymbol{\pi}(t) \sim \boldsymbol{\pi}_n(t)$, where $\boldsymbol{\pi}_n(t) = \{\pi_{n,k}(t) : k \geq 1\}$. Let $\mathbf{s}(t) = (s_n(t))_{n=1}^N$, and $\boldsymbol{\pi}(t) = \{\boldsymbol{\pi}_n(t) : n \in \{1, ..., N\}\}$. We consider $\boldsymbol{\pi}(t)$ to be generated by a DPMM, i.e., $\boldsymbol{\pi}_n(t) \sim \text{GEM}(\gamma)$, where GEM stands for the Griffiths, Engen and McCloskey stick breaking process and γ is a concentration hyperparameter [14]. From [15,16], we approximate the Dirichlet process with its degree K weak limit given by

$$\boldsymbol{\pi}_n(t) \sim \text{Dir}(\gamma/K, ..., \gamma/K), \tag{8}$$

where $\text{Dir}(\cdot)$ is the Dirichlet distribution. However, in view of Proposition 1, we need to introduce an amount of "stickiness" to $s_n(t)$ so that nodes do not change clusters frequently. Therefore, we use the following instead:

$$\pi_{n,k}(t) \mid s_n(t-1), \gamma, \kappa \sim \text{Dir}(p_1(t), p_2(t), ..., p_K(t)), \tag{9}$$

where for each $k \in \{1, 2, ..., K\}$,

$$p_k(t) = \begin{cases} \gamma/K + \kappa & \text{if } s_n(t-1) = k, \\ \gamma/K & \text{otherwise,} \end{cases}$$

where κ is a constant that controls the probability of a node changing to a different cluster at time t. We choose κ to be sufficiently large to induce a certain amount of stickiness in the node cluster index.

In summary, at each time $t \geq 1$, our model is the following:

$$\begin{aligned} s_n(t) \mid \boldsymbol{\pi}(t) &\sim \boldsymbol{\pi}_n(t), \\ \pi_{n,k}(t) \mid s_n(t-1), \gamma, \kappa &\sim \text{Dir}(p_1(t), ..., p_K(t)), \\ \theta_k(t) \mid \theta_k(t-1), \sigma, \lambda_k &\sim \mathcal{N}\left(\lambda_k \theta_k(t-1),\ \sigma^2\right), \\ y_n(t) | \boldsymbol{\theta}(t), s_n(t) = k, \alpha_n, \beta_n, \varsigma_n(t) &\sim \mathcal{N}\left(\alpha_n \theta_k(t) + \beta_n,\ \varsigma_n^2(t)\right). \end{aligned}$$

4. INFERENCE ALGORITHM

To perform inference using the dynamic Bayesian nonparametric model presented in Section 3, we use a block Gibbs sampling method based on the forward filtering backward sampling approach. Our algorithm is detailed in Algorithm 1.

Algorithm 1 Bayesian Inference Algorithm

Input: $\boldsymbol{\alpha}^{i-1}$, $\boldsymbol{\beta}^{i-1}$,$\boldsymbol{\theta}^{i-1}$,$\boldsymbol{\pi}^{i-1}$, $\mathbf{s}^{i-1}$ from previous iteration (iteration $i-1$) and data $\mathbf{y} = (\mathbf{y}(t))_{t=1}^T$
Output: new set of samples: $\boldsymbol{\alpha}^i$, $\boldsymbol{\beta}^i$,$\boldsymbol{\theta}^i$,$\boldsymbol{\pi}^i$ $\mathbf{s}^i$
  for $t = 1$ to T **do**
    Sample $\mathbf{s}^i(t)$ using (10) and $\boldsymbol{\pi}^i(t)$ using (11).
  end for
  Block sample $\boldsymbol{\theta}^i(1:T)$ using (12).
  Sample $\boldsymbol{\alpha}^i$ using (16).
  Sample $\boldsymbol{\beta}^i$ using (19).
  return $\boldsymbol{\alpha}^i$, $\boldsymbol{\beta}^i$,$\boldsymbol{\theta}^i$,$\boldsymbol{\pi}^i$ and $\mathbf{s}^i$

4.1. Sampling $\boldsymbol{\pi}^i(t)$ and $\mathbf{s}^i(t)$

At the i-th iteration, we sample $s_n(t)$ using the following posterior distribution, where $\boldsymbol{\pi}(t)$, $\boldsymbol{\theta}(t)$, $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ are set to their respective sample values from the $(i-1)$-th iteration:

$$\begin{aligned} &p\left(s_n(t) = k \mid y_n(t), \boldsymbol{\pi}(t), \boldsymbol{\theta}(t), \boldsymbol{\alpha}, \boldsymbol{\beta}\right) \\ &\propto p(s_n(t) = k \mid \boldsymbol{\pi}(t)) p(y_n(t) \mid s_n(t) = k, \boldsymbol{\theta}(t), \alpha_n, \beta_n) \\ &= \pi_{n,k}(t) p(y_n(t) \mid \theta_k(t), \alpha_n, \beta_n) \end{aligned} \tag{10}$$

Let $l_k(t) \triangleq \sum_{n=1}^N \delta(s_n(t) - k)$, where $\delta(\cdot)$ is the Dirac delta function, be the number of $s_n(t)$ sampled from (10) that takes value k. From (8), and the conjugacy of the Dirichlet prior, we have

$$\pi_{n,k}(t) \mid \mathbf{s}(t), s_n(t-1), \gamma, \kappa \sim \text{Dir}(p'_1(t), p'_2(t), ..., p'_K(t)), \tag{11}$$

where

$$p'_k(t) = \begin{cases} \gamma/K + l_k(t) + \kappa & \text{if } s_n(t-1) = k, \\ \gamma/K + l_k(t) & \text{otherwise,} \end{cases}$$

$p(y_n(t) \mid s_n(t) = k, \theta_k(t), \alpha_n, \beta_n, \varsigma_n(t))$ is from (1). We then sample $\pi_{n,k}(t)$ from (11).

4.2. Sampling $\boldsymbol{\theta}^i(1:T)$

We set $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ to their respective sample values from the $(i-1)$-th iteration. Let $\mathbf{y}_k(t) \triangleq \{y_n(t) \mid s_n(t) = k\}$ and $\boldsymbol{\psi} \triangleq \{m_k(0), v_k(0), \boldsymbol{\alpha}, \boldsymbol{\beta}, \sigma, \boldsymbol{\varsigma}, \lambda_k\}$. Then

$$\begin{aligned} &p(\theta_k(1:T) \mid \mathbf{y}_k(1:T), \boldsymbol{\psi}) \\ =&p(\theta_k(T) \mid \mathbf{y}_k(1:T), \boldsymbol{\psi}) \prod_{t=1}^{T-1} p(\theta_k(t) \mid \theta_k(t+1), \mathbf{y}_k(1:t), \boldsymbol{\psi}), \end{aligned} \tag{12}$$

where

$$\begin{aligned} &p(\theta_k(t) \mid \theta_k(t+1), \mathbf{y}_k(1:t), \boldsymbol{\psi}) \\ \propto &p(\theta_k(t+1) \mid \theta_k(t), \boldsymbol{\psi}) p(\theta_k(t) \mid \mathbf{y}_k(1:t), \boldsymbol{\psi}). \end{aligned} \tag{13}$$

To sample from (12), we need to first obtain $p(\theta_k(t) \mid \mathbf{y}_k(1:t), \boldsymbol{\psi})$, where $t \in \{1, 2, ..., T\}$. With the forward filtering method, we have

$$\theta_k(t) \mid \mathbf{y}_k(1:t), \boldsymbol{\psi} \sim \mathcal{N}\left(m_k(t),\ v_k(t)\right), \tag{14}$$

where

$$\begin{aligned} v_k(t) &= (\boldsymbol{\alpha}^T \boldsymbol{\Sigma}_k(t)^{-1} \boldsymbol{\alpha} + R_k(t)^{-1})^{-1}, \\ m_k(t) &= v_k(t) \left(-\boldsymbol{\alpha}^T \boldsymbol{\Sigma}_k(t)^{-1} \boldsymbol{\beta} + \boldsymbol{\alpha}^T \boldsymbol{\Sigma}_k(t)^{-1} \mathbf{y}_k(t) + \frac{a_k(t)}{R_k(t)} \right), \\ \boldsymbol{\Sigma}_k(t) &= \text{diag}(\{\varsigma_n(t) \mid s_n(t) = k\}), \\ a_k(t) &= \lambda_k m_k(t-1), R_k(t) = \lambda_k^2 v_k(t-1) + \sigma^2. \end{aligned}$$

Using (14) with t replaced by T, we obtain $p(\theta_k(T) \mid \mathbf{y}_k(1:T), \boldsymbol{\psi})$, from which we sample $\theta_k(T)$.

From (13), (14), and the conjugacy of the Gaussian prior, we have

$$p(\theta_k(t) \mid \theta_k(t+1), \mathbf{y}_k(1:t), \boldsymbol{\psi}) \sim \mathcal{N}\left(h_k(t),\ H_k(t)\right), \tag{15}$$

where

$$\begin{aligned} B_k(t) &= v_k(t) \lambda_k / R_k(t+1), \\ H_k(t) &= v_k(t) - B_k(t) R_k(t+1) B_k(t), \\ h_k(t) &= m_k(t) + B_k(t)(\theta_k(t+1) - a_k(t+1)). \end{aligned}$$

Using (15), we sample $\theta_k(t)$ conditioned on the sample for $\theta_k(t+1)$, starting from $t = T-1$.

4.3. Sampling $\boldsymbol{\alpha}^i$ and $\boldsymbol{\beta}^i$

We set $\boldsymbol{\beta}$ to their respective sample values from the $(i-1)$-th iteration. Let $\phi_n(t) \triangleq \theta_{s_n(t)}(t)$, then the posterior distribution of α_n is given by

$$\begin{aligned} &p(\alpha_n \mid \beta_n, \mathbf{y}_n(1:T), \boldsymbol{\phi}_n(1:T), \sigma_{1,n}, \varsigma_n(t)) \\ \propto &p(\mathbf{y}_n(1:T) \mid \alpha_n, \beta_n, \boldsymbol{\phi}_n(1:T), \varsigma_n(t)) p(\alpha_n \mid \sigma_{1,n}) \\ = &\prod_{t=1}^T p(y_n(t) \mid \alpha_n, \beta_n, \phi_n(t), \varsigma_n(t)) p(\alpha_n \mid \sigma_{1,n}). \end{aligned}$$

From (1) and (7), we obtain the following distribution:

$$\alpha_n \mid \beta_n, \mathbf{y}_n(1:T), \boldsymbol{\phi}_n(1:T), \sigma_{1,n}, \varsigma_n(t) \sim \mathcal{N}\left(M_n,\ v_n\right), \tag{16}$$

where

$$v_n = \frac{1}{2} \left[\sum_{t=1}^T \frac{\phi_n(t)^2}{2\varsigma_n^2(t)} + \frac{1}{2\sigma_{1,n}^2} \right]^{-1}, \tag{17}$$

$$M_n = v_n \left[\sum_{t=1}^T \frac{y_n(t) \phi_n(t)}{\varsigma_n^2(t)} + \frac{\mu_{1,n}}{\sigma_{1,n}^2} - \sum_{t=1}^T \frac{\beta_n \phi_n(t)}{\varsigma_n^2(t)} \right]. \tag{18}$$

Similarly,

$$\beta_n \mid \alpha_n, \mathbf{y}_n(1:T), \boldsymbol{\phi}_n(1:T), \sigma_{1,n}, \varsigma_n(t) \sim \mathcal{N}\left(T_n,\ G_n\right), \tag{19}$$

where

$$G_n = \frac{1}{2}\left[\sum_{t=1}^{T}\frac{1}{2\varsigma_n^2(t)} + \frac{1}{2\sigma_{2,n}^2}\right]^{-1}, \quad (20)$$

$$T_n = G_n\left[\sum_{t=1}^{T}\frac{y_n(t)}{\varsigma_n^2(t)} + \frac{\mu_{2,n}}{\sigma_{2,n}^2} - \sum_{t=1}^{T}\frac{\alpha_n\phi_n(t)}{\varsigma_n^2(t)}\right]. \quad (21)$$

5. SIMULATION RESULTS

We simulate $N = 40$ sensors observing 4 signals. At initialization, the signals are chosen to be $\theta_1(0) \sim \mathcal{N}(45,\ 1)$, $\theta_2(0) \sim \mathcal{N}(25,\ 1)$, $\theta_3(0) \sim \mathcal{N}(10,\ 1)$ and $\theta_4(0) \sim \mathcal{N}(-10,\ 1)$. The observation time period is $T = 20$. We use the following distributions to generate data:

$$\alpha_i \sim \mathcal{N}\left(1,\ 0.12^2\right),\ \beta_i \sim \mathcal{N}\left(0,\ 1.2^2\right),$$
$$y_n(t) = \alpha_n\theta_{s_n(t)}(t) + \beta_n + \omega_n(t),$$
$$\omega_n(t) \sim \mathcal{N}\left(0,\ \varsigma_n^2(t)\right),\ \varsigma_n^2(t) \sim \mathcal{N}\left(1,\ 0.1^2\right),$$
$$\epsilon_k(t) \sim \mathcal{N}\left(0,\ 2^2\right),\ \text{for all } k = 1, \ldots, 4.$$

In Algorithm 1,we set the following initial values: $K = 10$, $\{\lambda_k\}_{k=1}^{K}$=1, $\{\alpha_n = \mu_{1,n}, \beta_n = \mu_{2,n}\}_{n=1}^{N}$, $\{m_k(0)\}_{k=1}^{K}$={55, 45, 35, 25, 10, 0, -10, -25, -35, -45}. The estimated sensor gains and offsets are shown in Fig. 2.

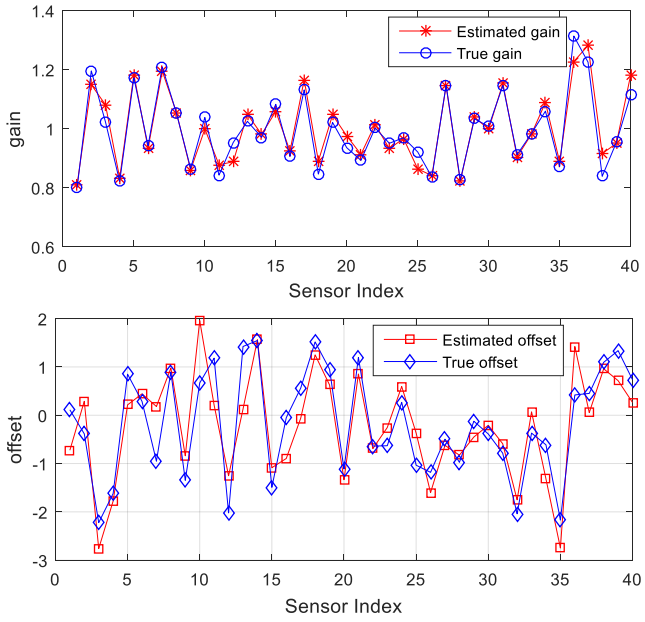

Fig. 2. Estimated sensor gains and offsets.

The average square error (ASE) is defined to be the square deviation of an estimated parameter from its true value, averaged over all the sensors. In our simulation, the ASE of the sensor gain estimation is 0.0019 and the ASE of the sensor offset estimation is 0.3362. In comparison, the prior variance of the sensor gains and offsets are 0.0144 and 1.44, respectively.

We compare our approach with following methods:

1. The gain and offset of each sensor are estimated according to its own observations using the maximum a posteriori (MAP) method.

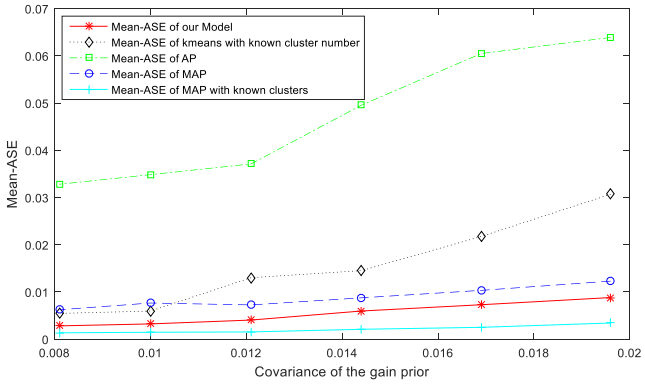

Fig. 3. Comparison of different methods for sensor gain estimation.

2. We assume that sensor clusters are known a priori, and sensor gains and offsets in the same cluster are estimated using MAP with all sensor observations from the same cluster. This represents the best possible performance one can get.
3. Assuming that the number of sensor clusters are known a priori, we apply k-means to first cluster the sensors based on the time series of sensor observations, and then MAP to estimate the sensor gains and offsets in each cluster separately.
4. We use affinity propagation (AP) [17] to perform clustering and then MAP estimation in each cluster separately.

In Fig. 3, we plot the mean ASE of the estimated sensor gains for the different methods against the prior variance of the Gaussian distribution used to generate the sensor gains. Comparing the performance of methods with and without a priori cluster information, we see that the correct clustering of the sensors can dramatically improve the estimation accuracy. We also observe that our method outperforms the methods that first perform clustering with k-means or AP and then conduct blind calibration. This is because both k-means and AP cluster the observed sensor measurements treating them as a vector, and do not utilize the evolution information of sensor signal.

6. CONCLUSION

In this paper, we have proposed a dynamic Bayesian nonparametric model for sensor blind calibration, where each sensor makes noisy observations of a signal that is evolving according to a first-order AR process. The number of underlying signals and the signals themselves are not known a priori. Our approach does not require several of the restrictive assumptions used in the literature for blind calibration. We also do not require a prior knowledge of which sensors are observing the same signal (sensors observing the same underlying signals are said to be in the same cluster). We have developed a block Gibbs sampling method to perform inference on our model, with simulation results suggesting that our performance is better than methods that first perform clustering with k-means or affinity propagation and then conduct blind calibration.

7. ACKNOWLEDGMENT

This research is supported in part by the Singapore Ministry of Education Academic Research Fund Tier 2 grant MOE2014-T2-1-028, and the Delta-NTU Corporate Lab through the National Research Foundation (NRF) corporate lab@university scheme.

8. REFERENCES

[1] L. Balzano and R. Nowak, "Blind calibration of sensor networks," in *ACM/IEEE Int. Conf. Inform. Process. Sensor Networks*, 2007, pp. 79–88.

[2] S. Guo, H. Zhang, Z. Zhong, J. Chen, Q. Cao, and T. He, "Detecting faulty nodes with data errors for wireless sensor networks," *ACM Trans. Sensor Networks*, vol. 10, no. 3, pp. 40, 2014.

[3] W. P. Tay, J. N. Tsitsiklis, and M. Z. Win, "On the impact of node failures and unreliable communications in dense sensor networks," *IEEE Trans. Signal Process.*, vol. 56, no. 6, pp. 2535–2546, 2008.

[4] W. P. Tay, J. N. Tsitsiklis, and M. Z. Win, "Bayesian detection in bounded height tree networks," *IEEE Trans. Signal Process.*, vol. 57, no. 10, pp. 4042–4051, 2009.

[5] K. Whitehouse and D. Culler, "Macro-calibration in sensor/actuator networks," *Mobile Netw. Applicat.*, vol. 8, no. 4, pp. 463–472, 2003.

[6] R. Tan, G. Xing, Z. Yuan, X. Liu, and J. Yao, "System-level calibration for fusion-based wireless sensor networks," in *IEEE Real-Time Syst. Symp.*, 2010, pp. 215–224.

[7] Y. Wang, A. Yang, Z. Li, X. Chen, P. Wang, and H. Yang, "Blind drift calibration of sensor networks using sparse Bayesian learning," *IEEE Sensors J.*, vol. 16, no. 16, pp. 6249–6260, 2016.

[8] M. S. Stanković, S. S. Stanković, and K. H. Johansson, "Distributed blind calibration in lossy sensor networks via output synchronization," *IEEE Trans. Autom. Control*, vol. 60, no. 12, pp. 3257–3262, 2015.

[9] B. Lee, S. Son, and K. Kang, "A blind calibration scheme exploiting mutual calibration relationships for a dense mobile sensor network," *IEEE Sensors J.*, vol. 14, no. 5, pp. 1518–1526, 2014.

[10] V. Bychkovskiy, S. Megerian, D. Estrin, and M. Potkonjak, "A collaborative approach to in-place sensor calibration," in *ACM/IEEE Int. Conf. Inform. Process. Sensor Networks*, 2003, pp. 301–316.

[11] K. P. Murphy, *Machine Learning: A Probabilistic Perspective*, Cambridge, MA: MIT Press, 2012.

[12] D. Kumar, S. Rajasegarar, and M. Palaniswami, "Automatic sensor drift detection and correction using spatial kriging and Kalman filtering," in *IEEE Int. Conf. Distributed Computing in Sensor Systems*, 2013, pp. 183–190.

[13] N. Ramakrishnan, E. Ertin, and R. L. Moses, "Gossip-based algorithm for joint signature estimation and node calibration in sensor networks," *IEEE J. Sel. Topics Signal Process.*, vol. 5, no. 4, pp. 665–673, 2011.

[14] Y. W. Teh, "Dirichlet process," in *Encyclopedia of Machine Learning*, pp. 280–287. London: Springer, 2011.

[15] H. Ishwaran and M. Zarepour, "Exact and approximate sum representations for the Dirichlet process," *Canadian J. Stat.*, vol. 30, no. 2, pp. 269–283, 2002.

[16] E. B. Fox, E. B. Sudderth, M. I. Jordan, and A. S. Willsky, "A sticky HDP-HMM with application to speaker diarization," *Ann. Appl. Stat.*, pp. 1020–1056, 2011.

[17] B. J. Frey and D. Dueck, "Clustering by passing messages between data points," *Science*, vol. 315, no. 5814, pp. 972–976, 2007.