Delft University of Technology

Delft Center for Systems and Control

Technical report 12-023

Identification of distributed-parameter systems with missing data*

Z. Hidayat, A. Núñez, R. Babuška, and B. De Schutter

If you want to cite this report, please use the following reference instead:
Z. Hidayat, A. Núñez, R. Babuška, and B. De Schutter, "Identification of distributed-parameter systems with missing data," *Proceedings of the 2012 IEEE International Conference on Control Applications*, Dubrovnik, Croatia, pp. 1014–1019, Oct. 2012.

Delft Center for Systems and Control
Delft University of Technology
Mekelweg 2, 2628 CD Delft
The Netherlands
phone: +31-15-278.24.73 (secretary)
URL: `https://www.dcsc.tudelft.nl`

*This report can also be downloaded via `https://pub.deschutter.info/abs/12_023.html`

Identification of Distributed-Parameter Systems with Missing Data

Z. Hidayat A. Núñez R. Babuška B. De Schutter

Abstract— **In this paper we address the identification of linear distributed-parameter systems with missing data. This setting is relevant in, for instance, sensor networks, where data are frequently lost due to transmission errors. We consider an identification problem where the only information available about the system are the input-output measurements from a set of sensors placed at known fixed locations in the distributed-parameter system. The model is represented as a set of coupled multi-input, single-output autoregressive with exogenous input (ARX) submodels. Total least-squares estimation is employed to obtain an unbiased parameter estimate in the presence of sensor noise. The missing samples are reconstructed with the help of an iterative algorithm. To approximate the value of the variables of interest in locations with no sensors, we use cubic B-splines to preserve the continuity of the first-order and second-order spatial derivatives. The method is applied to a simulated one-dimensional heat-conduction process.**

I. Introduction

Advances in the microelectronics technology have enabled the development of wireless sensor networks with applications in a variety of fields including industrial process control, greenhouse climate monitoring and control, traffic management systems, etc. An advantage of wireless sensor networks is the reduction of the clutter of wires in the case of large and complex processes with many measurement points. However, the use of wireless sensors is susceptible to the following problems:

- Intermittent measurements, where the cause of missing data can be faults in sensors or in the signal transmission.
- Asynchronous sampling due the absence of globally synchronized clocks.
- Different noise levels at different sensor locations.

These problems cause difficulties in control-related tasks, such as system identification and state estimation, which use input-output measurements to model the process. In this case, it is necessary to develop approaches that are able to deal with the above issues. In this paper we consider the identification of linear distributed-parameter dynamic systems with missing data.

For lumped-parameter systems, methods of system identification with missing data can be found in the literature. These methods generally extend similar approaches known from statistics. Isaksson [1] presented a method based on expectation maximization and used a Kalman filter to reconstruct the missing data for the ARX model structure. Raghavan *et al.* [2] proposed another expectation-maximization approach to identify state-space models with irregular output sampling, while Ding and Ding [3] developed a recursive least-squares approach using outputs of auxiliary models to substitute the missing data.

Several techniques for the identification of distributed-parameter systems with unknown structure and parameters have been proposed [4], assuming that the data are complete. One of the methods is the identification based on a Green's function[1]. This method has been used by, e.g., Gay and Ray [5] who applied the singular value decomposition to estimate a time-invariant Green function. Zheng *et al.* [6] combined the singular value decomposition and the Karhunen-Loève expansion to estimate a time-varying Green function, and Doumanidis and Fuorligkas [7] used a spatially discretized Green function estimation for a thermal processing system. The disadvantage of the Green function approximation is that the Green function is estimated only for a given working point. Therefore, the function has to be approximated again for different working points.

Another approach is based on a finite-difference method that converts a partial differential equation into a set of difference equations with unknown parameters by partitioning the spatial variables using a grid [8]. Each of the difference equations corresponds to a grid point. The unknown parameters are computed using optimization methods that minimize an error equation criterion. Voss *et al.* [9] proposed a parameter estimation method based on the alternating conditional expectation for a pre-selected structure. Guo and Billings [10] presented the use of implicit integration to form the unknown algebraic equations and used optimized polynomial basis functions to estimate the structure and parameters.

The discretization by the finite-difference method can result in a class of lattice dynamical systems[2] [4]. Identification of such systems has been investigated by Parlitz and Merkwirth [11] and Mandelj *et al.* [12] who used parametric statistical methods for parameter estimation. Orthogonal forward regression was used to estimate the parameters of a nonlinear autoregressive model with exogenous inputs, as proposed by Guo and Billings [13] and Coca and Billings [14]. The main drawback of the finite-difference approaches is the high model order and high complexity for nonlinear systems. Interested readers may refer to [4] for a recent survey on the identification of distributed-parameter systems.

The authors are with Delft Center for Systems and Control, Delft University of Technology, Mekelweg 2, 2628 CD Delft, The Netherlands Email: {z.hidayat, a.a.nunezvicencio, r.babuska, b.deschutter}@tudelft.nl

[1]Green's functions are a class of kernel functions used to solve inhomogeneous differential equations subject to specific initial conditions or boundary conditions.

[2]A lattice dynamical system consists of subsystems that are arranged as a lattice and each of the subsystems is coupled to nearby subsystems.

To the best knowledge of the authors, the identification methods for distributed-parameter systems in the literature assume that the data are complete. Therefore in this paper, an identification method for linear distributed-parameter systems with missing data is proposed. The method uses an iterative scheme to impute the missing data based on a multivariable prediction model identified from measurements of neighboring sensors. A cubic B-spline-based function approximator is used to obtain predictions in the spatial locations where no sensors are present.

The rest of the paper is organized as follows: Section II formulates the identification problem of linear distributed-parameter systems with missing data. In Section III, the proposed identification method is presented. Simulations and analysis are given in Section IV, and Section V concludes the paper.

II. Problem Formulation

Consider a distributed-parameter system described by a linear partial differential equation with an unknown structure and parameters. For simplicity and without loss of generality, we assume a 1D first-order temporal partial differential equation. Higher-order systems can be tackled similarly. The system is described as follows:

$$\frac{\partial g(z,t)}{\partial t} = f\left(g(z,t), \frac{\partial g(z,t)}{\partial z}, \cdots, \frac{\partial^n g(z,t)}{\partial z^n}, u(z,t), w(z,t)\right) \tag{1}$$

where $g(z,t)$ is the variable of interest, $z \in \mathscr{Z} \subset \mathbb{R}$ is the spatial coordinate variable, n is the highest order of the spatial derivative, $w(t)$ is the process noise, and $u(z,t)$ is the input vector. The input $\boldsymbol{u}(t)$ and the noisy output vector of the process $\boldsymbol{y}(t) = \boldsymbol{g}(t) + \boldsymbol{v}(t)$ are defined at discrete locations:

$$\boldsymbol{u}(t) = \begin{bmatrix} u(z_1,t) \\ \vdots \\ u(z_{N_\mathrm{u}},t) \end{bmatrix}, \boldsymbol{g}(t) = \begin{bmatrix} g(z_1,t) \\ \vdots \\ g(z_{N_\mathrm{s}},t) \end{bmatrix}, \boldsymbol{v}(t) = \begin{bmatrix} v(z_1,t) \\ \vdots \\ v(z_{N_\mathrm{s}},t) \end{bmatrix}$$

where z_i is the location coordinate of the i-th sensor, N_u is the number of inputs, N_s is the number of sensors, and $\boldsymbol{v}$ is the measurement noise vector. When a measurement is not available from sensor i, the corresponding element of $\boldsymbol{y}(t)$ is assigned a special value, such as NaN (not a number). The boundary conditions at the set of boundaries $\mathscr{Z}_\mathrm{b}$ are

$$h\left(g(z,t), \frac{\partial g(z,t)}{\partial (z)}, \cdots, \frac{\partial^{n-1} g(z,t)}{\partial z^{n-1}}\right) = 0, \quad \forall z \in \mathscr{Z}_\mathrm{b}, \forall t \tag{2}$$

and the initial condition is

$$g(z,0) = g_0(z), \quad \forall z \in \mathscr{Z}. \tag{3}$$

Given the discrete spatial locations of the sensors and inputs and the sampling period T_s, the measurement data from (1) are sampled for every $t = kT_\mathrm{s}, k \in \mathbb{N} \cup \{0\}$ from every sensor. In the sequel, we denote the discrete time instant with k to simplify the notation. The spatial domain is discretized at grid points $z_\mathrm{g} \in M_\mathrm{g}$, where M_g is the set of grid point coordinates. Sensors and point inputs are placed in locations $z_\mathrm{s} \in M_\mathrm{s}$ and $z_\mathrm{u} \in M_\mathrm{u}$ respectively, where $M_\mathrm{s} \subset M_\mathrm{g}$ is the set of sensor locations and $M_\mathrm{u} \subset M_\mathrm{g}$ is the set of input locations. Note that the grid spacing is not necessarily uniform, but it always coincides with the location of the sensors and inputs.

For the sake of identification, we assume that the only available information about the system are input-output measurements of variables obtained through sensors placed at specific locations in the system. However, due to faults in the sensors or in the data transmission, some samples are missing randomly. In addition, we have missing data in space, because the measurements are only available at the specific sensor locations. These measurements represent partial information that we have about the value of the distributed variable in space.

To illustrate this setting, Figure 1 shows the available and missing measurements at discrete time steps k. Grid points are shown as the intersections between the horizontal dashed lines and the discrete time step vertical lines, whereas sensors/actuators are placed at the labeled grid points. The available measurements are marked by circles, while the missing measurements are marked by squares.

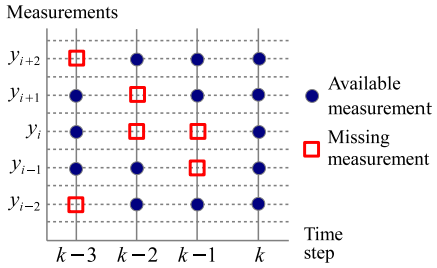

Fig. 1. Missing data in the measurements.

The goal is to construct a model of the underlying distributed-parameter system by using the incomplete data.

III. Proposed Identification Method

We start with a view that the solution of (1) at a particular discrete time instant k is a manifold (a curve in 1D) in the space approximated by the values at the corresponding grid points. The measurements available at time instant k can be seen as spatial samples of the solution and will be used to compute the approximation of the curve at the grid points where no sensors are placed. At the same time, the measurements at sensor i can be seen as temporal signals and they can be modeled using system identification. The spatial and temporal view of the measurements complement each other and for the identification both views can be used sequentially or simultaneously. In this paper, we take the sequential approach, i.e., we start from the temporal identification to model measurements in the sensors and continue with the spatial approximation.

A. System identification with missing data

A common approach to identify a system when the input-output measurements are not complete is to reconstruct the missing data first. Data reconstruction methods are

presented, e.g., in [1]–[3]. In this paper, the missing data are initially extrapolated from the available measurements of the previous step, using the zero-order-hold principle (assuming for simplicity and without loss of generality that the first measurement is available). In Figure 1, the missing values $y_i(k-2)$ and $y_i(k-1)$ are imputed from the value of the previous measurement, $y_i(k-3)$. Other methods can be applied at this step as well. In the sequel these initial reconstructed values will be improved iteratively.

Once the missing data have been imputed, we can proceed to identify the model of $y_i(k)$. Each sensor measurement is predicted using a multiple-input, single-output (MISO) linear ARX model with the neighboring measurements as inputs. There are two ways to determine which neighboring inputs to use: 1) based on prior knowledge about the system, i.e., the knowledge of the spatial order of the PDE (10); 2) by using a suitable input selection method [15]. In this paper the neighbors are determined using prior knowledge.

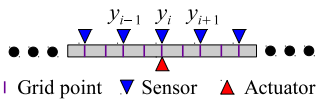

Fig. 2. An example of sensors/actuator configuration

For the ease of notation, we consider two neighboring sensors to sensor i, sensor $i-1$ and sensor $i+1$ according the configuration shown in Figure 2. However, the approach can be adapted to other configurations. Define the following regression vectors for the ARX model:

$$\begin{aligned}\boldsymbol{\varphi}_i^\top(k) &= \begin{bmatrix}\varphi_{y,i}^\top(k) & \varphi_{u,i}^\top(k)\end{bmatrix}\\ \varphi_{y,i}^\top(k) &= \begin{bmatrix}y_i(k) & \dots & y_i(k-m) & y_{i-1}(k) & \dots\\ & & y_{i-1}(k-m) & y_{i+1}(k) & \dots & y_{i+1}(k-m)\end{bmatrix}\\ \varphi_{u,i}^\top(k) &= \begin{bmatrix}u_i(k) & \dots & u_i(k-m)\end{bmatrix}\end{aligned} \tag{4}$$

where $y_i(k)$ and $u_i(k)$ are the output and the input at grid point i, respectively, $\top$ denotes the matrix transpose, and m is the order of the model. The model order m can be determined by using prior knowledge about the process or the Akaike information criterion [15]. Without loss of generality, we take the same order for the input and output polynomials. The prediction of model $y_i(k)$ is written as:

$$\hat{y}_i(k) = \boldsymbol{\varphi}_i^\top(k-1)\boldsymbol{\theta}_i \tag{5}$$

where $\boldsymbol{\theta}_i$ is the parameter vector of model i. This vector is estimated as the total least-squares approach [16] solution of the following set of linear equations

$$\boldsymbol{\Phi}_i\boldsymbol{\theta}_i = \boldsymbol{Y}_i \tag{6}$$

where $\boldsymbol{\Phi}_i$ and $\boldsymbol{Y}_i$ contain the measurements in their rows:

$$\boldsymbol{\Phi}_i = \begin{bmatrix}\boldsymbol{\varphi}_i^\top(1)\\ \vdots\\ \boldsymbol{\varphi}_i^\top(N)\end{bmatrix}, \quad \boldsymbol{Y}_i = \begin{bmatrix}y_i(1)\\ \vdots\\ y_i(N)\end{bmatrix} \tag{7}$$

for N the number of samples. A brief overview on the total-least squares is presented in Appendix I.

In the first iteration, i.e., after identifying all the models in the spatial domain, we simulate the each model to get the estimates of the measurements and replace the initial estimates of the missing data with the output of the simulations

$$y_i^j(k) = \begin{cases}\boldsymbol{\varphi}_i^\top(k-1)\boldsymbol{\theta}_i^j & \text{if } y_i(k) \text{ missing}\\ y_i(k) & \text{otherwise}\end{cases} \tag{8}$$

where the superscript j is the iteration index and $y_i^j(k)$ the measurement $y_i(k)$ at iteration j. The identification, the simulation, and the missing data estimates replacement are repeated again until the estimates of the parameters converge. As $y_i(k)$ is then available for all sensors, in which the missing samples are replaced by $\hat{y}_i(k)$, the spatial function approximation can be performed next. We denote by $\tilde{\boldsymbol{y}}(k)$ the vector containing the measurements and the imputed values.

The identification step of the proposed method is outlined in Algorithm 1. Note that this algorithms does not only produce the estimate of the parameter vector $\boldsymbol{\theta}_i$, $i=1,\dots,N_\mathrm{s}$, but also a modified data set $\{\tilde{\boldsymbol{y}}(k)\}_{k=1}^N$ in which the missing samples are replaced by the model predictions.

Algorithm 1 Temporal identification with missing data

Input: $\{\boldsymbol{u}(k),\boldsymbol{y}(k)\}_{k=1}^N, \varepsilon$
  $j \leftarrow 1$
  for each step k **and** each sensor i **do**
    $y_i^j(k) \leftarrow \begin{cases}y_i(k-1) & \text{if } y_i(k) \text{ is missing}\\ y_i(k) & \text{otherwise}\end{cases}$
  end for
  repeat
    for each sensor i **do**
      Form $\boldsymbol{\Phi}_i$ and $\boldsymbol{Y}_i$ according to (7)
      Compute $\boldsymbol{\theta}_i^j$ using total least-squares method (15)
    end for
    for each step k **and** sensor i **do**
      $y_i^j(k) \leftarrow \begin{cases}\boldsymbol{\varphi}_i^\top(k-1)\boldsymbol{\theta}_i^j & \text{if } y_i(k) \text{ missing}\\ y_i(k) & \text{otherwise}\end{cases}$
    end for
    $j \leftarrow j+1$
  until $\|\boldsymbol{\theta}_i^j - \boldsymbol{\theta}_i^{j-1}\| < \varepsilon$
  $\tilde{\boldsymbol{y}}(k) = \boldsymbol{y}^j(k) \quad \forall k$
Output: $\boldsymbol{\theta}_i$, $i=1,\dots,N_\mathrm{s}$, $\{\tilde{\boldsymbol{y}}(k)\}_{k=1}^N$

The proposed method is an off-line method yielding a model that can be used for monitoring and prediction. However, the method can easily be extended to an on-line method by using recursive identification techniques.

B. Spatial approximation with splines

This is the last step of the algorithm. As mentioned in the previous section, we can view the grid point locations that are not measured as missing samples in space. This is the dual view of the missing data for distributed-parameter systems. The available methods in system identification allow us to use them to model the temporal data with missing samples and for the missing spatial data we can use function approximators.

Let the solution of (1) at a discrete time instant k be approximated by a suitable function approximator υ as:

$$\hat{g}(z,k) = \upsilon\Big(z, y_1(k), \ldots, y_{N_\mathrm{s}}(k)\Big) \tag{9}$$

where $\hat{g}(z,k)$ is the approximated solution curve, and $y_i(k)$ the measurements. When the measurement $y_i(k)$ is not available, we can use its estimate that has been calculated from the temporal model (8).

Consider a heated bar as an example for Figure 2, where sensors are placed to measure the temperature. Plotting the measurements as a function of discrete-time k results in Figure 1 if there are missing samples in the measurement data. The unlabeled horizontal dashed lines are grid points that have no sensors and whose temperature values are approximated using a function approximator.

There are different function approximators that can be used to approximate the unmeasured grid points, for instance, splines. For the heated bar we can use, e.g., cubic B-splines. The cubic B-splines fulfill the conditions of the solution of the 1D heat conduction equation: the heat function and its first and second derivatives are continuous [17]. For other cases, other function approximators can be used. A brief overview on splines as function approximators is presented in the appendix. The spatial approximation step is presented in Algorithm 2.

Algorithm 2 Spatial approximation

Input: $\{\tilde{y}_i(k)\}_{k=1}^{N}$, for $i = 1, \ldots, N_\mathrm{s}$, z

Compute B-spline model according (16) in the appendix for $\tilde{y}_i(k)$

Calculate coefficients of the B-spline to minimize (19)

Compute $\hat{g}(z,k)$ using (20)

Output: $\hat{g}(z,k)$

IV. Simulation Example

A one-dimensional heat conduction process is taken as an example to illustrate the effectiveness of the proposed approach. The equation of the process is:

$$\frac{\partial T(z,t)}{\partial t} = \frac{1}{\rho C_\mathrm{p}}\left[\kappa \frac{\partial^2 T(z,t)}{\partial z^2}\right] \tag{10a}$$

$$T(0,t) = T_{\mathrm{b},1} \qquad T(L,t) = T_{\mathrm{b},2} \tag{10b}$$

$$T(z,0) = T_0 \tag{10c}$$

where T is the temperature of the rod, ρ the density of the rod, C_p the heat capacity, κ the thermal conductivity, and z the spatial coordinate of length. Equations (10b) and (10c) are the boundary conditions and the initial condition respectively. The rod's parameters are listed in Table I.

In order to generate the data for our simulation, we proceed as follows: the partial-differential equation (10a) is discretized in space, by using the central approximation of the second-order spatial derivative:

$$\left.\frac{\partial^2 g(z,t)}{\partial z^2}\right|_{z=i} \approx \frac{g_{i+1}(t) - 2g_i(t) + g_{i-1}(t)}{(\Delta_z)^2} \tag{11}$$

TABLE I

Rod parameters

Parameters	Values	Units
ρ	8700	$\mathrm{kg\,m^{-3}}$
κ	400	$\mathrm{W\,m^{-1}\,K^{-1}}$
C_p	385	$\mathrm{J\,kg^{-1}\,K^{-1}}$
L	0.6	m
T_0	35	°C
T_s	1	s
Δ_z	0.02	m

with Δ_z the spatial discretization interval. The spatial discretization results in an equidistant grid. Applying (11) to (10a) and simplifying the notation with respect to the grid-point index, (10a) becomes the following ordinary partial differential equation:

$$\frac{\mathrm{d}T_i(t)}{\mathrm{d}t} = C_\mathrm{z} T_{i-1}(t) - 2C_\mathrm{z} T_i(t) + C_\mathrm{z} T_{i+1}(t) \tag{12}$$

with i the grid point index and

$$C_\mathrm{z} = \frac{\kappa}{\rho C_\mathrm{p} \Delta_z^2}\,.$$

Model (12) is also discretized temporally to get the model that is used to generate the simulation data.

The 1D heat conduction equation is simulated for a bar with length 0.6 m. The bar is divided into 31 grid points and the first and the last grid points are boundary conditions, which are used as inputs. A number of 3000 measurements from each sensor are generated using the discretized model.

The missing measurements are chosen randomly and independently for each sensor and the events of the missing measurements are generated with random permutation using Matlab function `randperm`. The simulations use four sets measurement data, each of which has a different percentage of missing measurements: 10% (data set 1), 20% (data set 2), 30% (data set 3), and 40% (data set 4). We arrange the data sets such that the percentage of missing measurements is increasing and such that the data set with a larger percentage of missing sample includes missing samples from the data set with a smaller percentage of missing samples. This way, we can see the influence of the increase of the missing samples consistently.

From all grid points, it is assumed that temperature at nodes 4, 7, 10, 13, 16, 19, 22, 25, and 28 is measured and corrupted with Gaussian noise of mean value 0 and standard deviation 0.3. The sensor at node 4 is denoted as sensor 1, the sensor at node 7 as sensor 2, and so on. Nodes 1 and 31 are boundary conditions. System (10) does not have inputs; however, we vary the values of the boundary conditions randomly such that they act as inputs. It is assumed that the boundary condition data are complete.

The temperature at sensor i is approximated using an ARX model. The orders of the models can be determined by following the model determination method in system identification. In our case, we set the order of 1 for the input polynomials and the output polynomials, because we know from the prior knowledge that the system is a first order

system. We use the adjacent measurements as input since we know that the temperature at a grid point depends on the measurements at the adjacent grid points (see (12)).

First, we analyze the convergence of the parameter estimate throughout the iterations of Algorithm 1. Figure 3 shows the difference $\|\hat{\theta}_{c,i} - \hat{\theta}_i^j\|_2$ between the parameters estimated from the incomplete data and from the complete data plotted against the iteration number for sensors 1, 3, 5, 7, and 9. Here, $\hat{\theta}_{c,i}$ denotes the parameter vector estimated from the complete data at location z_i, and j is the iteration number. We can see that the parameters estimated from the incomplete data converge to the parameters estimated from the complete data already after the second iteration for all different percentages of missing data. It can also be seen that in the first iteration, the parameter deviations increase as the percentage of the missing sample increases.

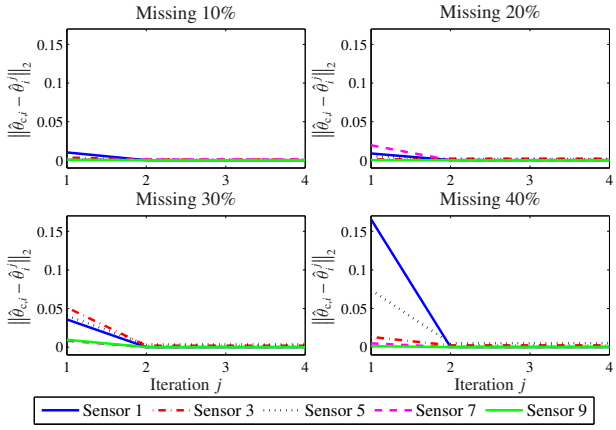

Fig. 3. The parameters estimated from the incomplete data converge rapidly within three iterations of Algorithm 1.

Figure 4 shows snapshots of the simulated spatial profile using the model identified from the complete data and from the incomplete data for 20% and 40% of missing samples. The outputs of the temporal model at the sensor locations are used to compute the spline, using Matlab function `spap2`, which provides a least-squares estimate of the B-spline parameters. It can be seen that the estimates are close to the measurements.

V. Conclusions and Future Work

A method of identification of linear distributed-parameter systems with missing data has been presented in this paper. The method is based on a spatiotemporal discretization of a partial differential equation. The spatial domain discretization uses a finite-difference method to represent the values of the variables at grid points in space while the temporal domain uses zero-order hold discretization. Measurements are assumed at some specific grid points. In the method, we do system identification with missing data to get models in the temporal domain. We subsequently use a least-squares spline function approximator to estimate the unmeasured values of the variables on grid points in the spatial domain. In a simulation with a 1D heat conduction equation we have

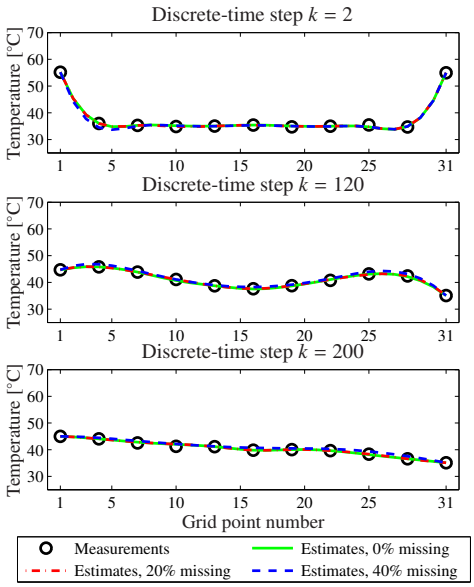

Fig. 4. Snapshots of the spatiotemporal estimation at the whole grid at discrete time step 2, 120, and 200.

shown that the obtained model works well. The estimates obtained from the model are able to follow the measurement data.

The proposed method is a general framework. There are several extensions that can be considered. Currently we have dealt with a simple linear problem and as the next step we will consider nonlinear systems. The use of the finite-difference method requires a relative large number of sensors at the grid points. Therefore, it will be useful to find a lower bound of the number of sensors and also the sensor locations to obtain a sufficiently accurate model. In this paper we considered the uniform grid. The use of non-uniform grids is also an important topic for investigation.

VI. Acknowledgment

This research is partially funded by the European Union Seventh Framework Programme [FP7/2007-2013] under grant agreement no. 257462 HYCON2 Network of Excellence (HYCON2)". The first author gratefully acknowledges the support of the Government of the Republic of Indonesia, Ministry of Communication and Information Technology.

References

[1] A. Isaksson, "Identification of ARX-models subject to missing data," *IEEE Transactions on Automatic Control*, vol. 38, no. 5, pp. 813–819, May 1993.

[2] H. Raghavan, A. Tangirala, R. Gopaluni, and S. Shah, "Identification of chemical processes with irregular output sampling," *Control Engineering Practice*, vol. 14, no. 5, pp. 467– 480, 2006.

[3] F. Ding and J. Ding, "Least-squares parameter estimation for systems with irregularly missing data," *International Journal of Adaptive Control and Signal Processing*, vol. 24, no. 7, pp. 540–553, 2009.

[4] H. Li and C. Qi, "Modeling of distributed parameter systems for applications – a synthesized review from time-space separation," *Journal of Process Control*, vol. 20, no. 8, pp. 891–901, 2010.

[5] D. H. Gay and W. H. Ray, "Identification and control of distributed parameter systems by means of the singular value decomposition," *Chemical Engineering Science*, vol. 50, no. 10, pp. 1519–1539, 1995.

[6] D. Zheng, K. Hoo, and M. Piovoso, "Finite dimensional modeling and control of distributed parameter systems," in *Proceedings of the 2002 American Control Conference (ACC '02)*, vol. 6, Anchorage, AK, USA, Jun. 2002, pp. 4377–4382.
[7] C. Doumanidis and N. Fourligkas, "Temperature distribution control in scanned thermal processing of thin circular parts," *IEEE Transactions on Control Systems Technology*, vol. 9, no. 5, pp. 708–717, Sep. 2001.
[8] D. Coca and S. A. Billings, "Direct parameter identification of distributed parameter systems," *International Journal of Systems Science*, vol. 31, no. 1, pp. 11–17, 2000.
[9] H. Voss, M. J. Bünner, and M. Abel, "Identification of continuous, spatiotemporal systems," *Physical Review E*, vol. 57, no. 3, pp. 2820–2823, Mar. 1998.
[10] L. Guo and S. Billings, "Identification of partial differential equation models for continuous spatio-temporal dynamical systems," *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 53, no. 8, pp. 657–661, Aug. 2006.
[11] U. Parlitz and C. Merkwirth, "Prediction of spatiotemporal time series based on reconstructed local states," *Physical Review Letters*, vol. 84, no. 9, pp. 1890–1893, Feb. 2000.
[12] S. Mandelj, I. Grabec, and E. Govekar, "Statistical approach to modeling of spatiotemporal dynamics," *International Journal of Bifurcation and Chaos*, vol. 11, no. 11, pp. 2731–2738, 2001.
[13] L. Guo and S. Billings, "State-space reconstruction and spatio-temporal prediction of lattice dynamical systems," *IEEE Transactions on Automatic Control*, vol. 52, no. 4, pp. 622–632, Apr. 2007.
[14] D. Coca and S. Billings, "Identification of coupled map lattice models of complex spatio-temporal patterns," *Physics Letters A*, vol. 287, no. 1-2, pp. 65–73, 2001.
[15] L. Ljung, *System Identification: Theory for the User*, 2nd ed. Prentice Hall, 1999.
[16] Å. Björck, *Numerical Methods for Least Squares Problems*. SIAM, 1996.
[17] P. Dierckx, *Curve and Surface Fitting with Splines*. Oxford University Press, 1993.
[18] J. Ramos and J.-F. Durand, "Identification of nonlinear systems using a B-splines parametric subspace approach," in *Proceedings of the 1999 American Control Conference*, vol. 6, San Diego, CA, USA, 1999, pp. 3955–3959.

APPENDIX I
TOTAL LEAST-SQUARES METHOD [16]

The total least-squares is an approach of solving a set of N linear equations:

$$a_1 x_{j1} + \cdots + a_m x_{jm} = y_j \tag{13}$$

where $a_i, i = 1,\ldots,m$ are the parameters and $x_{ji}, j = 1,\ldots,N$ and y_j are known. In matrix form, (13) can be written as

$$\boldsymbol{\Phi}\boldsymbol{\theta} = \boldsymbol{Y} \tag{14}$$

where

$$\boldsymbol{\Phi} = \begin{bmatrix} x_{11} & \cdots & x_{1m} \\ \vdots & \ddots & \vdots \\ x_{N1} & \cdots & x_{Nm} \end{bmatrix}, \quad \boldsymbol{\theta} = \begin{bmatrix} a_1 \\ \vdots \\ a_m \end{bmatrix}, \quad \boldsymbol{Y} = \begin{bmatrix} y_1 \\ \vdots \\ y_N \end{bmatrix}$$

Assuming that there are errors in both sides of (14), $(\boldsymbol{\Phi} + \Delta\boldsymbol{\Phi})\boldsymbol{\theta} = \boldsymbol{Y} + \Delta\boldsymbol{Y}$, the total least-squares method solve the following optimization problem:

$$\begin{aligned} &\min_{\Delta\boldsymbol{\Phi},\Delta\boldsymbol{Y}} \|(\Delta\boldsymbol{\Phi}, \Delta\boldsymbol{Y})\|_{\mathrm{F}} \\ &\text{s.t.} \quad (\boldsymbol{\Phi} + \Delta\boldsymbol{\Phi})\boldsymbol{\theta} = \boldsymbol{Y} + \Delta\boldsymbol{Y} \end{aligned} \tag{15}$$

where $\|\cdot\|_{\mathrm{F}}$ is the Frobenius norm. The Frobenius norm of matrix A is defined as $\sqrt{\mathrm{tr}\{AA^*\}}$, for the superscript * denotes the complex conjugate operation of a matrix.

APPENDIX II
LINEAR LEAST-SQUARES WITH SPLINES [17]

A spline function $s(\cdot)$ defined on interval an $[a,b]$ consists of connected piecewise polynomials, each of which is defined on an interval $[\lambda_i, \lambda_{i+1}]$ with $\lambda_i < \lambda_{i+1}, i = 0,\ldots,r; (\lambda_0 = a, \lambda_{r+1} = b)$ for a strictly increasing sequence λ_i, called knots. A spline $s(\cdot)$ has degree $p > 0$ if it satisfies:

1) Between each knot interval $[\lambda_i, \lambda_{i+1}]$, $s(\cdot)$ is given by a polynomial that has a maximum degree p
2) $s(\cdot)$ and its derivatives up to order $p-1$ are continuous on $[a,b]$

B-splines are a class of splines that have been used for identification of nonlinear systems in, e.g., [18]. A B-spline $D_{i,p+1}$ of degree p whose knots are $\lambda_i,\ldots,\lambda_{i+p+1}$ can be expressed as the following recursion

$$\begin{aligned} D_{i,p+1}(l) &= \frac{l-\lambda_i}{\lambda_{i+p}-\lambda_i} D_{i,p}(l) + \frac{\lambda_{i+p+1}-l}{\lambda_{i+p+1}-\lambda_{i+1}} D_{i+1,p}(l), \\ D_{i,1}(l) &= \begin{cases} 1 & \text{if } l \in [\lambda_i, \lambda_{i+1}) \\ 0 & \text{if } l \notin [\lambda_i, \lambda_{i+1}) \end{cases} \end{aligned} \tag{16}$$

Any spline $s(\cdot)$ can be expressed as a linear combination of B-splines. Given a set of knots $\lambda_i, i = 0,\ldots,r+1$, a number of $r-p+1$ linearly independent B-splines of degree p can be constructed, i.e., $D_{i,p+1}, i = 0,\ldots,r-p$ [17].

Using the B-splines $D_{i,p+1}(l)$, any spline $s(\cdot)$ has a unique representation:

$$s(l) = \sum_{i=-p}^{r} c_i D_{i,p+1}(l) \tag{17}$$

where c_i are the B-spline coefficients of $s(\cdot)$. The basis has a sum of unity property on $[a,b]$:

$$\sum_{i=-p}^{r} D_{i,p+1}(l) = 1, \quad \forall l \in [a,b] \tag{18}$$

Given a set of data points $(l_j, d_j), j = 1,\ldots,N$, with $a \le l_j \le l_{j+1} \le b$, we can formulate a least-squares problem to fit the spline to the data. In other words, we want to find a spline $s(\cdot)$ of degree p on $[a,b]$ with given knots $\lambda_i, i = 0,\ldots,r+1 (\lambda_0 = a, \lambda_j = b)$ such that the cost function

$$\mathscr{J}_{\mathrm{S}} = \sum_{i=1}^{N} \left(d_j - \sum_{n=-p}^{r} c_i D_{i,p+1}(l_j) \right)^2 \tag{19}$$

is minimized using linear least-squares. The estimated value at l_j can be calculated as

$$\hat{d}_j = \sum_{i=-p}^{r} c_i D_{i,p+1}(l_j) \tag{20}$$