Identification of Quadratic in-the-State System Using Nonlinear Programming

Wael SULEIMAN and André MONIN
LAAS - CNRS, 7 avenue du Colonel Roche, 31077 Toulouse Cedex 4
France

Abstract—The identification of the Quadratic in-the-State Systems (QSS) is the objective of this paper. This class, on its own, enjoys a useful model for many nonlinear dynamic systems. Moreover, it represents a bilinear system in a feedback loop. In order to identify QSS, we propose to minimize the output error with respect to the system’s parameters using a local gradient search. In order to reduce the amount of gradient search to the minimal value, we propose an efficient implementation of the algorithm by projecting out the directions that do not change the output error function. Furthermore, we extend the proposed method to deal with multiple short experiments.

I. INTRODUCTION

During the last years, the attention of researchers has been pointed to the identification of nonlinear systems. Such attention is the natural result of the intrinsic limits of the linear systems in many fields of sciences such as ecology, physiology, biology and socio-economics [1], [2], [3], [4]. In fact, the interest of linear systems is due to their simple analytical properties. However, dealing with complex systems which are multivariable and as well nonlinear has pushed the identification theory and the efforts of researchers are focused to overcome the analytical difficulties of nonlinear systems.

Consider the following nonlinear finite dimensional system

\[ x_{t+1} = f(x_t, u_t) \]
\[ y_t = h(x_t, u_t) + v_t \]

(1)

where \( u_t \in \mathbb{R}^m \) is the input signals, \( y_t \in \mathbb{R}^p \) is the output signals and the measurement noise \( v_t \) is assumed to be a white-noise that is independent of the input signal and with zero mean. In the literature, we find that the bilinear systems is one of the classic models [5], [6], [7] which is proposed to approximate the nonlinear system (1). This class of systems is linear separately with respect to the state \( (x_t) \) and the input signal \( (u_t) \), and can be characterized by the following model

\[ x_{t+1} = A_b x_t + B_b u_t + N_b (u_t \otimes x_t) \]
\[ y_t = C_b x_t + v_t \]

(2)

where \( \otimes \) denotes the Kronecker product. It is proven that this class of systems is dense, in \( L^2 \) sense, in the class of analytical nonlinear systems. For that, studying the properties and the identification of bilinear systems has been subject of active research during the last decade [8], [9], [10].

In this paper, we will consider the identification of bilinear system in a feedback loop which is presented in Fig. 1.

![Fig. 1. Quadratic system as a bilinear system in a feedback loop.](image)

Using the property \( FG \otimes HJ = (F \otimes H)(G \otimes J) \), the representation of the equivalent model is given by

\[ x_{t+1} = Ax_t + Bu_t + N(u_t \otimes x_t) + Q(x_t \otimes x_t) \]
\[ y_t = Cx_t + v_t \]

(3)

where

\[ A = A_b - B_b K \]
\[ B = B_b H \]
\[ N = N_b (H \otimes I_b) \]
\[ Q = -N_b (K \otimes I_b) \]

(4)

The class of systems given by the model (3) is called Quadratic in-the-State Systems (QSS). In fact, this class enjoys a useful model more than a representation of a bilinear system in a feedback loop. For example, in ecology and biology where the population dynamics of two interacting species may be described by the prey-predator Volterra equation [11], [12], which has the exactly structure of QSS. In the engineering area, there are also several examples of QSS, e.g. the dynamics of a general AC machine [13]. Note that, the realization (3) is not unique. Assume the following transformation of the state

\[ z_t = T^{-1} x_t \]

(5)

where \( T \in \mathbb{R}^{n \times n} \) is a nonsingular matrix. Then, we obtain the following structure

\[ z_{t+1} = \tilde{A} z_t + \tilde{B} u_t + \tilde{N} (u_t \otimes z_t) + \tilde{Q} (z_t \otimes z_t) \]
\[ y_t = \tilde{C} z_t + v_t \]

(6)

where

\[ \tilde{A} = T^{-1} AT \]
\[ \tilde{B} = T^{-1} B \]
\[ \tilde{N} = T^{-1} N (I_m \otimes T) \]
\[ \tilde{Q} = T^{-1} Q (T \otimes T) \]
\[ \tilde{C} = CT \]

(7)

As a consequence, the relation input/output does not change by transforming the state \( x_t \) according to (5).
II. IDENTIFICATION PROCEDURE

The identification of Structure (3) relies upon the estimation of the matrices \( A, B, C, N, Q \). Without any assumption on the structure of these matrices, we assume that all matrices are fully parameterized. Let us define

\[
\theta = \begin{bmatrix}
\text{vec}(A) \\
\text{vec}(B) \\
\text{vec}(N) \\
\text{vec}(Q) \\
\text{vec}(C)
\end{bmatrix}
\] (8)

where \( \text{vec}(\cdot) \) denotes the vectorization operator defined as follows

\[
\text{vec}(M) \in \mathbb{R}_{m \times n} \rightarrow \mathbb{R}_{mn}
\]

\[
\text{vec}(M) = \text{vec} \left[ m_1 \ m_2 \cdots m_n \right] = [m_1^T \ m_2^T \cdots m_n^T]^T
\]

Suppose that we have an estimation \( \hat{\theta} \) of \( \theta \), and a set of input/output data from the real system \( \{u_i, y_i : i = 0, 1, \cdots, L\} \). The estimated output \( \hat{y}_i (\hat{\theta}) \) can be given as follows

\[
\hat{y}_i (\hat{\theta}) = A (\hat{\theta}) \hat{x}_i + B(\hat{\theta}) u_i + N(\hat{\theta}) (u_i \otimes \hat{x}_i) + Q(\hat{\theta}) (\hat{x}_i \otimes \hat{x}_i)
\] (9)

our goal is achieved if \( \hat{y}_i (\hat{\theta}) \) approximates the output \( y_i \) accurately enough. This criterion can be transformed into the minimization of the output error with respect to the parameter \( \theta \) which leads to an output-error identification problem. Considering the following output-error cost function

\[
J_L(\theta) = \frac{1}{L} \sum_{k=1}^{L} \| y_k - \hat{y}_k (\theta) \|^2_2 = \frac{1}{L} E_L(\theta)^T E_L(\theta)
\] (10)

where

\[
E_L(\theta) = [e(1)^T \ e(2)^T \cdots e(L)^T]^T
\]

is the error vector in which \( e(k) = y_k - \hat{y}_k (\theta) \). The minimization of (10) is a nonlinear, nonconvex optimization problem. The numerical solution of this problem can be calculated by different algorithms. In the literature of output-error identification methods, we find that the gradient search method is the popular adopted method. This iterative method is based on the updating of the system parameters \( \theta \) as follows

\[
\theta^{i+1} = \theta^i - (\psi_L(\theta^i) + \lambda^{i+1} I)^{-1} \psi_L(\theta^i) E_L(\theta^i)
\] (12)

Where \( \lambda^{i+1} \) is the regularization parameter and

\[
\psi_L(\theta) \triangleq \frac{\partial E_L(\theta)}{\partial \theta^T}
\] (13)

is the Jacobian of the error vector \( E_L(\theta) \).

A. Computing the iterative parameter update

In order to compute the update rule (12), the following quantities \( E_L(\theta) \) and \( \psi_L(\theta) \) must be computed. Computing the vector \( E_L(\theta) \) can be done by simulating the system (9) that corresponds to \( \theta^{i+1} \). Note that this simulation brings out the state \( \hat{x}_i \) and \( \hat{y}_i \). In order to simulate \( \psi_L(\theta^{i+1}) \), we should compute the derivative of \( \hat{y}_i \) with respect to \( \theta^{i+1} \). Let us define

\[
\zeta_j = \frac{\partial \hat{y}_i}{\partial \theta_j}
\] (14)

where \( \theta_j \) is the \( j^{th} \) element of the vector \( \theta \). The computation of \( \frac{\partial \hat{y}_i}{\partial \theta_j} = \left[ \frac{\partial \hat{y}_i}{\partial \theta_1}, \cdots, \frac{\partial \hat{y}_i}{\partial \theta_l} \right]^T \), where \( l \) is the number of parameters in \( \theta \), can be made using the following model

\[
\zeta_{i+1} = \Lambda \zeta_i + \frac{\partial A}{\partial \theta_j} \hat{x}_i + \frac{\partial B}{\partial \theta_j} u_i + N(u_i \otimes \zeta_i) + \frac{\partial N}{\partial \theta_j} (u_i \otimes \hat{x}_i) + Q(\zeta_i \otimes \hat{x}_i) + \frac{\partial Q}{\partial \theta_j} (\hat{x}_i \otimes \hat{x}_i)
\]

\[
\frac{\partial \hat{y}_i}{\partial \theta_j} = C \zeta_j
\] (15)

III. LOCAL PARAMETERIZATION

As we mentioned the structure (3) is not unique, as consequence the minimization of \( J_L(\theta) \) does not have a unique solution. Indeed, the optimal solution can be made unique by choosing a suitable parameterization. Such parameterization for QSS does not exist up to our knowledge. However, a local parameterization is proposed for the bilinear systems [14], where the directions that do not change the cost function \( J_L(\theta) \) are identified and projected out at each iteration, for that only the active parameters are updated. Analogously to previous method, we will develop a local parameterization of QSS.

Recall that two realizations of QSS in the state space representation are similar if their coefficient matrices are related by Equation (7), where the transformation matrix \( T \) parameterizes the subset of equivalent models. Note that, this subset defines a manifold. In order to identify the tangent plane of the manifold, we linearize the relation (7) around the identity matrix \( T = I_n \). Considering a small perturbation \( T = I_n + \Delta T \), by using the approximation \((I_n + \Delta T)^{-1} \approx I_n - \Delta T\) and neglecting all second-order terms, we obtain

\[
\tilde{A} = A - \Delta T A + A \Delta T
\]

\[
\tilde{B} = B - \Delta T B
\]

\[
\tilde{N} = N - \Delta T N + N(I_n \otimes \Delta T)
\]

\[
\tilde{Q} = Q - \Delta T Q + Q(I_n \otimes \Delta T) + Q(\Delta T \otimes I_n)
\]

\[
\tilde{C} = C + \Delta T T
\]
where

$$\theta = \begin{bmatrix} \text{vec}(A) \\ \text{vec}(B) \\ \text{vec}(N) \\ \text{vec}(Q) \\ \text{vec}(C) \end{bmatrix}$$

and

$$\hat{\theta} = \begin{bmatrix} \text{vec}(\hat{A}) \\ \text{vec}(\hat{B}) \\ \text{vec}(\hat{N}) \\ \text{vec}(\hat{Q}) \\ \text{vec}(\hat{C}) \end{bmatrix}$$

Then

$$\hat{\theta} = \theta + \begin{bmatrix} \text{vec}(-\Delta T A + \Delta T) \\ \text{vec}(-\Delta T B) \\ \text{vec}(-\Delta T N + N (I_n \otimes \Delta T)) \\ \text{vec}(-\Delta T Q + Q (I_n \otimes \Delta T) + Q (\Delta T \otimes I_n)) \\ \text{vec}(\Gamma) \end{bmatrix}$$

By defining

$$\Pi_{i,K} = \begin{bmatrix} 0_{n \times (i-1)n} & I_n & 0_{n \times (K-i)n} \end{bmatrix}$$

it is possible to write $I_n \otimes \Delta T$ as follows

$$I_n \otimes \Delta T = \sum_{i=1}^{m} \sum_{j=1}^{n} \Pi_{i,j} \Pi_{i,m} \Delta T (i,j)$$

On the other hand, $\Delta T \otimes I_n$ can be written as follows

$$\Delta T \otimes I_n = \sum_{i=1}^{m} \sum_{j=1}^{n} \Pi_{i,j} \Pi_{i,m} \Delta T (i,j)$$

where $\Delta T (i,j)$ is the element of the row $i$ and column $j$ of $\Delta T$. Using the property $\text{vec}(ABC) = (C^T \otimes A) \text{vec}(B)$, we obtain

$$\text{vec}(Q (\Delta T \otimes I_n)) = \sum_{i=1}^{n} \sum_{j=1}^{n} (\Pi_{i,j} \otimes Q) \text{vec}(\Pi_{j,m} \Delta T (i,j))$$

$$= \Gamma_Q \text{vec}(\Delta T)$$

where

$$\Gamma_Q = [\gamma_{1,1} \gamma_{1,2} \cdots \gamma_{1,n} \gamma_{2,1} \gamma_{2,2} \cdots \gamma_{2,n} \cdots \gamma_{n,n}]$$

and

$$\gamma_{j,i} = (\Pi_{i,m} \otimes Q) \text{vec}(\Pi_{j,m})$$

Using analogous logic, the relation between $\theta$ and $\hat{\theta}$ becomes

$$\hat{\theta} = \theta + M_{\theta} \text{vec}(\Delta T)$$

where

$$M_{\theta} = \begin{bmatrix} -A^T \otimes I_n + I_n \otimes A \\ -B^T \otimes I_n \\ -N^T \otimes I_n + \sum_{i=1}^{m} \Pi_{i,m} \otimes (N \Pi_{i,m}) \\ -Q^T \otimes I_n + \sum_{i=1}^{m} \Pi_{i,m} \otimes (Q \Pi_{i,m}) + \Gamma_Q \\ I_n \otimes C \end{bmatrix}$$

Lemma 1: The left null space of $M_{\theta}$ (25) contains the directions in which the parameters should be modified to lead a change in the value of cost function $J_N(\theta)$ (10).

Proof:
The equation (24) shows that the tangent space of the manifold of all systems similar to $(A,B,N,Q,C)$ is equal to the column space of the matrix $M_{\theta}$ (25). Since the left null space of the matrix $M_{\theta}$ is orthogonal complement to the column space, the directions in which the value of the cost function (10) changes are those related to the left null space of $M_{\theta}$.

Note that, the left null space of $M_{\theta}$ can be efficiently obtained by a QR decomposition

$$M_{\theta} = \begin{bmatrix} \mathcal{D}_1 & \mathcal{D}_2 \end{bmatrix} \begin{bmatrix} \mathbf{R} \\ \mathbf{0} \end{bmatrix}$$

Then a basis of the left null space of $M_{\theta}$ is $\mathcal{D}_2$. The new update rule becomes

$$\theta^t = \theta^{t-1} - \mathcal{D}_2 (\mathcal{D}_2^T \psi_L \psi_L \mathcal{D}_2 + \lambda I)^{-1} \mathcal{D}_2^T \psi_L E_t$$

where $\mathcal{D}_2$ and $\psi_L$ depend on $\theta^{t-1}$. Note that since $\mathcal{D}_2$ depends on the past parameter $\theta^{t-1}$ the QR (26) must be computed at each iteration.

A. Summary of the implementation algorithm

We can resume the algorithm of implementation as follows

1) Calculate the state $\hat{x}_t$ and $\hat{y}_t$ by simulating the system (9) with $\theta = \theta^{t-1}$.
2) Compute $E_t(\theta)$ using (11).
3) Calculate the matrix $M_{\theta}$ using (25).
4) Calculate the QR decomposition of $M_{\theta}$ (26), from which we obtain $\mathcal{D}_2$.
5) Calculate the matrix $\psi_L$ using (15), we suppose that \{ $\zeta_{j,i} = 0$, \( j = 1,2, \cdots , l \}$.
6) Calculate the update rule of the gradient search algorithm using (27).
7) Perform the termination test for minimization. If true, the algorithm stops. Otherwise, return to Step 1. i.e. compute the values $J_L(\theta^{t-1})$ and $J_L(\theta^t)$ using (10) and test if $\| J_L(\theta^t) - J_L(\theta^{t-1}) \|_2$ is small enough.

IV. DEALING WITH MULTIPLE DATA SETS

In the purpose of identifying practical applications, we apply various sets of input. The objective of each set is exciting one or more modes of the system. In real experiments, sometimes we obtain short sets of input/output data due to a large time sampling period or short tracked phenomena. In these cases, we should exploit all the data sets which we have to obtain an accurate model of the system. Moreover, in the case of short experiments the effect of initial conditions can not be neglected, so we should estimate them. Assume that we have the sets \{ $u_i, y_i : t = 1,2, \cdots , l$, and $i = 1,2, \cdots , q$ \}.

If we consider the initial conditions, the vector of...
parameters \( \theta \) becomes
\[
\theta = \begin{bmatrix}
\text{vec}(A) \\
\text{vec}(B) \\
\text{vec}(Q) \\
\text{vec}(C) \\
x_0 \\
x_0 \\
\vdots \\
x_0 
\end{bmatrix}
\] (28)

where \( x_0 \) is the initial value of the state corresponded to \( j^{th} \) data set. The estimated output \( \hat{y}_j^{(t)}(\hat{\theta}) \) of the data set number \( j \) is given by the following model
\[
\hat{y}_j^{(t+1)} = A(\hat{\theta})\hat{y}_j^{(t)} + B(\hat{\theta})u_j^{(t)} + N(\hat{\theta}) (u_j^{(t)} \circ \hat{x}_j^{(t)}) + Q(\hat{\theta}) \left( \hat{x}_j^{(t)} \circ \hat{\theta} \right)
\] (29)

The output error function for all data sets can be done as follows
\[
J_p(\theta) = \frac{1}{q} \sum_{j=1}^{q} \sum_{k=1}^{L_j} \left\| y_j^{(t)} - \hat{y}_j^{(t)}(\theta) \right\|^2 = \frac{1}{q} E_q(\theta)^T E_q(\theta)
\] (30)

where
\[
E_q(\theta) = \begin{bmatrix}
E_{t_1}^1(\theta)^T \\
E_{t_2}^2(\theta)^T \\
\vdots \\
E_{t_q}^q(\theta)^T
\end{bmatrix}
\] (31)

and
\[
E_{t_i}^i(\theta) = \frac{1}{\sqrt{L_i}} \left[ e_i(1)^T e_i(2)^T \cdots e_i(L_i)^T \right]^T
\] (32)

is the error vector in which \( e_i(k) = y_i^k - \hat{y}_i^k(\theta) \). The minimization of (30) can be calculated by using the gradient search method as follows
\[
\theta^{t+1} = \theta^t - (\psi_q^T(\theta^t)\psi_q(\theta^t) + \lambda^t I)^{-1} \psi_q^T(\theta^t) E_q(\theta^t)
\] (33)

where
\[
\psi_q(\theta) = \begin{bmatrix}
\psi_{t_1}^1(\theta) \\
\psi_{t_2}^2(\theta) \\
\vdots \\
\psi_{t_q}^q(\theta)
\end{bmatrix}
\] (34)

and
\[
\psi_{t_i}^i(\theta) = \frac{\partial E_{t_i}^i(\theta)}{\partial \theta^T}
\] (35)

A. Local parameterization

Similar to the case of a single experiment, we will look for a local parameterization in order to define the direction in which the cost function (30) is invariant.

Recall that, two realization of QSS are similar and their vectors of parameters are \( \theta \) and \( \hat{\theta} \) respectively.
\[
\theta = \begin{bmatrix}
\text{vec}(A) \\
\text{vec}(B) \\
\text{vec}(Q) \\
\text{vec}(C) \\
x_0 \\
\vdots \\
x_0
\end{bmatrix}
\] \( \text{and} \)
\[
\hat{\theta} = \begin{bmatrix}
\text{vec}(\hat{A}) \\
\text{vec}(\hat{B}) \\
\text{vec}(\hat{Q}) \\
\text{vec}(\hat{C}) \\
x_0 \\
\vdots \\
x_0
\end{bmatrix}
\] (36)

By considering a small perturbation \( T = I_n + \Delta T \) and analogously to the case of a single experiment we obtain
\[
\hat{\theta} = \theta + M_\theta(\Delta T)
\] (37)

where
\[
M_\theta = \begin{bmatrix}
-A^T \otimes I_n + I_n \otimes A \\
-B^T \otimes I_n \\
-N^T \otimes I_n + \sum_{i=1}^{m} \Pi^T_i \otimes (N \Pi^T_i m) \\
-Q^T \otimes I_n + \sum_{i=1}^{n} \Pi^T_i \otimes (Q \Pi^T_i m) + \Gamma_Q \\
-(x_0)^T \otimes I_n
\end{bmatrix}
\] (38)

then, we calculate the QR decomposition of \( M_\theta \) similar to (26), from which we obtain \( \Delta \). The new update rule becomes
\[
\theta^k = \theta^{k-1} - \Delta \left( \psi_q^T \psi_q \Delta + \lambda^k I \right)^{-1} \psi_q^T E_q
\] (39)

B. Computing the iterative parameter update

In order to compute the update rule (27), the following quantities \( E_q(\theta) \) and \( \psi_q(\theta) \) must be computed. For that, we simulate the systems (29) that corresponds to \( \theta^{k-1} \). Note that this simulation brings out the state \( \hat{x}_k \) and \( \hat{y}_k \) for \( i = 1, 2, \ldots, q \). In order to simulate \( \psi_q(\theta^{k-1}) \), we should compute the derivative of \( \{ \hat{y}_i : i = 1, 2, \ldots, q \} \) with respect to \( \theta^{k-1} \). Let us define
\[
\frac{\partial \psi_q^i}{\partial \theta_j} = \frac{\partial \psi_q^i}{\partial \theta_j}
\] (40)

where \( \theta_j \) is the \( j^{th} \) element of the vector \( \theta \). The computation of \( \frac{\partial \psi_q^i}{\partial \theta_j} \) can be made using the following model
\[
\frac{\partial \psi_q^i}{\partial \theta_j} = A \frac{\partial \psi_q^i}{\partial \theta_j} + \frac{\partial A}{\partial \theta_j} \psi_q^i + \frac{\partial B}{\partial \theta_j} u_j + \frac{\partial N}{\partial \theta_j} (u_j \circ \hat{x}_j)
\] (41)

Note that, as the initial condition is included into the vector of parameters then \( \frac{\partial \psi_q^i}{\partial \theta_j} = \frac{\partial \psi_q^i}{\partial \theta_j} \neq 0 \).
V. COMPUTING AN INITIAL ESTIMATION

The complexity of the iterative gradient search algorithm and the stability of the obtained structure depends on the choice of an initial estimation of the vector of parameters \( \theta \). One could first estimate a stable bilinear model, then use the matrices \( A, B, N, C \) of the model as initial guesses, and the matrix \( Q \) can be initialized as zero matrix.

In the case of multiple experiments, we use a single data set to calculate an initial estimation of the vector of parameters \( \theta \). Then by using this model we can estimate the initial conditions of each data set.

VI. ILLUSTRATIVE EXAMPLES

In this section, first we identify a QSS using a single experiment. Second we point out that the proposed algorithm to deal with multiple short data sets provides a model more precisely than treating each experiment separately.

We define the Model accuracy as the Percent Variance Accounted For (%VAF)

\[
%\text{VAF} = \left( 1 - \frac{\sum_{t=1}^{L} (y_t - \hat{y}_t)^2}{\sum_{t=1}^{L} (y_t - \bar{y})^2} \right) \times 100
\]

where \( \hat{y}_t \) denotes the estimated output signal and \( \bar{y} \) is the mean value of \( y_t \).

A. Identifying QSS using single experiment

In this case, we consider that the length of the input and output signals is \( L = 1200 \) samples, and we have used \( T_{id} = \frac{L}{2} \) samples for the identification purpose, and the rest of them \( T_{val} = L - T_{id} = \frac{L}{2} \) samples for the validation purpose. We consider the following example

\[
A = \begin{pmatrix} 0.6 & 0 \\ 0 & 0.4 \end{pmatrix}, \quad B = \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix}, \quad C = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \\
N = \begin{pmatrix} 0.7 & 0 & 0.3 \\ 0 & 0.2 & 0.4 \end{pmatrix}, \quad Q = \begin{pmatrix} -0.4 & 0 & -0.4 \\ -0.5 & 0 & -0.3 \end{pmatrix}
\]

The input is two-dimensional uniform white noise and the standard deviation of independent noises \( (\nu_t) \) is equal to 0.1. The initial estimated model is a bilinear system which is obtained by using the Matlab package CUESID [15]. We consider the output signal without the measurement noise when we calculate the accuracy of the identified models in validation task. The accuracies reported in Table I show that the proposed method has efficiently identified the QSS structure.

To deal with a clear graphical representation, we represent the data over a window of 150 samples in Fig. 2.

B. Identifying QSS using multiple experiments

In this case, we consider that we have executed 10 experiments \( (q = 10) \) with random initial conditions. That means we have obtained the sets \( \{ u_i, y_i, x_i \}_{i=1}^{10} \) and we suppose that \( L_1 = L_2 = L_3 = L_4 = L_5 = 200 \) and \( L_6 = L_7 = L_8 = L_9 = L_{10} = 100 \). The input for each experiment is an uniform white noise and the standard deviation of independent noises \( (\nu_t) \) is equal to 0.1. We consider the same example (42) of the single experiment case.

In order to show that the obtained model by considering all data sets is more accurate than that obtained by using a single experiment, we have proceeded as follows. First, we have used a single experiment (the first one) to identify the model of the system and without considering the estimation of initial conditions. Second, we have used \( q - 1 = 9 \) data sets to identify the model of system and the initial conditions.

<table>
<thead>
<tr>
<th>Identification task</th>
<th>Accuracy (%VAF)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Using single data set</td>
<td>Using multiple data sets</td>
</tr>
<tr>
<td>First output</td>
<td>73.6</td>
</tr>
<tr>
<td>Second output</td>
<td>75.8</td>
</tr>
<tr>
<td>Validation task</td>
<td></td>
</tr>
<tr>
<td>First output</td>
<td>74.1</td>
</tr>
<tr>
<td>Second output</td>
<td>76.8</td>
</tr>
</tbody>
</table>

The validation of the two models has been done on the rest one of data sets (Fig. 3). Note that, we suppose that the initial conditions of the data set used for validation are known. It is clear from Table II that in this case of short experiments, using multiple experiments leads to a more accurate model.

VII. CONCLUSION

In this paper, we have proposed a method to identify the Quadratic in-the-State Systems (QSS). The method is based on a local parameterization of the state-space representation of the QSS and subsequent gradient search in the resulting local parameter space. A bilinear model is used to initialize the optimization task, in the reported examples this procedure seems to work properly. The algorithm has successfully applied to identify the QSS in the case of single experiment as well as the case of multiple short experiments. This case is infrequently treated in the literature of identification theory.
Fig. 2. The real outputs without noise (solid line) with the outputs of the identified QSS (dashed line) and bilinear system (dash-dotted line) and their errors are superimposed.

Fig. 3. The real outputs without noise (solid line) with the outputs of the identified QSS using all data sets (dashed line) and the identified QSS using a single data set (dash-dotted line) and their errors are superimposed.

REFERENCES