2011 50th IEEE Conference on Decision and Control and
European Control Conference (CDC-ECC)
Orlando, FL, USA, December 12-15, 2011

Set-membership identification of ARX models with quantized measurements

Marco Casini, Andrea Garulli, Antonio Vicino

***Abstract*— System identification with binary or quantized measurements is a problem relevant to a number of applications in different fields. While identification of FIR models has been studied in depth, more complex model structures still need to be investigated. In this paper, identification of ARX models with quantized measurements is addressed in a set membership setting. In particular, the problem of characterizing and bounding the feasible parameter set (FPS), i.e., the set of model parameters which are compatible with the available data, is tackled. Being the FPS in general nonconvex, an algorithm is proposed for constructing an outer approximation. The proposed technique relies on quasiconvex relaxations of the original problem, based on generalized linear fractional programming. Structural properties of the FPS and convergence issues are analyzed, and numerical examples are presented to validate the proposed procedure.**

I. Introduction

System identification in presence of binary valued or quantized measurements has received an increasing attention since the seminal work by Wang, Zhang and Yin [1], which introduced a general framework dealing with both stochastic and deterministic uncertainty representations. Binary data is generated by binary sensors which are widespread devices characterized by a threshold according to which the output is digitized. Similarly, multi-threshold sensors or banks cascaded of binary sensors generate quantized measurements, with a quantization error depending on the number of thresholds or binary sensors.

Motivation for this identification approach can be found in several engineering areas. Communication systems show contexts like ATM networks where traffic information, e.g., bit rate, queue length, is measured through binary sensors characterized by appropriate thresholds. Typical sensors used in monitoring and control systems of industrial production plants are binary devices, popular examples being chemical process sensors in gas and oil industry, or sensors monitoring liquid or pressure levels. Binary valued or quantized measurements can be easily found in a number of automotive applications, including switching sensors for exhaust gas oxygen, shift-by-wire and ignition systems, photoelectric sensors for position detection, Hall-effect sensors for speed and acceleration measurement, ABS, and others.

In recent years, a number of relevant issues have been addressed in the stochastic setting, including optimal input design and time complexity [1], consistency analysis of weighted least squares estimators [2], identification of Wiener systems [3], identification and tracking of linear systems with time-varying parameters [4], Expectation Maximization identification of FIR systems [5], design of state observers [6] (see [7] for a complete treatment and an extensive reference list for identification with either binary or quantized data). Statistical quantization theory is exploited in [8] to compute the likelihood function and the Cramer-Rao bound for models with quantized measurements.

When an unknown-but-bounded description of uncertainty is adopted, the identification problem is naturally formulated in the set-membership framework (see e.g., [9], [10]). Results on input design and time complexity for FIR systems, in presence of binary or quantized measurements, have been given in [11]–[13], while a framework taking into account the features of both the stochastic and the deterministic settings has been introduced in [14]. In this context, a key concept is represented by the so-called *feasible parameter set*, which is the set of all the model parameters that are compatible with the available information. While this problem is well understood for FIR models, for which the feasible set is a convex polytope, it has not been addressed so far for more complex model structures such as ARX. The main difficulty in the latter case comes from the fact that the nonlinearity present in the sensor, generates feasible sets which in general are nonconvex sets with nonlinear boundaries. The problem is similar to the one arising in the set-membership framework when dealing with errors-in-variables identification. In this context, an approach based on LMI relaxations has been proposed in [15], [16].

In this paper, we present an approach for bounding the feasible parameter sets of ARX models, for set-membership identification with quantized measurements. For given N input/output pairs and a priori information on the model parameters, the aim of this work is to construct an outer box approximation of the feasible set. The proposed technique relies on quasiconvex relaxations of the original bounding problem, based on generalized linear fractional programming [17]. An iterative algorithm is devised to progressively reduce the model parameter uncertainty set estimate.

The paper is organized as follows. Section II introduces notation and problem formulation. In Section III the problem of computing an outer box approximation of the feasible set is tackled, while in Section IV an algorithm showing faster convergence is proposed. In Section V, some structural properties of the feasible set in presence of binary measurements are analyzed. Numerical examples are presented in Section VI, while concluding remarks and future perspectives are reported in Section VII.

Authors are with the Dipartimento di Ingegneria dell'Informazione and the Centro per lo Studio dei Sistemi Complessi, Università di Siena, 53100 Siena, Italy (email: casini@ing.unisi.it; garulli@ing.unisi.it; vicino@ing.unisi.it).

II. Problem formulation

Let $\mathbb{R}^N$ denote the N-dimensional Euclidean space. A sequence of real numbers $\{x(t), t = 1, \ldots, N\}$ is identified by a vector $x \in \mathbb{R}^N$.

Let us consider an ARX SISO linear time-invariant model

$$y(t) = \sum_{i=1}^{n} a_i\, y(t-i) + \sum_{j=1}^{m} b_j\, u(t-j+1) + d(t) \quad (1)$$

where $y(t)$ is the model output, $u(t)$ is the input signal and $d(t)$ denotes the equation error. The disturbance $d(t)$ is assumed to be bounded by a known quantity, i.e., $|d(t)| \leq \delta,\ t = 1, 2, \ldots$. Observations at the system output are taken by a multi-valued sensor with $P \geq 1$ known thresholds $C_1, \ldots, C_P$, such that

$$s(t) = \sigma(y(t)) \triangleq \begin{cases} 0 & \text{if } C_0 < y(t) \leq C_1 \\ 1 & \text{if } C_1 < y(t) \leq C_2 \\ \vdots & \\ P & \text{if } C_P < y(t) \leq C_{P+1} \end{cases} \quad (2)$$

where $C_0 \triangleq -\infty$, $C_{P+1} \triangleq +\infty$. The special case $P = 1$ refers to a binary sensor, where the only information given by a measurement is $y(t) \leq C_1$ or $y(t) > C_1$.

Let $\theta = [a_1, \ldots, a_n, b_1, \ldots, b_m]' \in \mathbb{R}^{n+m}$ denote the ARX parameter vector and $\phi(t) = [y(t-1), \ldots, y(t-n), u(t), \ldots, u(t-m+1)]'$ the regressor vector. Then, (1) can be expressed in the standard regression form

$$y(t) = \phi'(t)\,\theta + d(t). \quad (3)$$

Let Θ_0 represent the prior information available on the ARX parameter vector. In this paper, it is assumed that Θ_0 is a box, i.e., $\theta \in \Theta_0$ means $a_i \in [\underline{a}_i, \overline{a}_i]$, $i = 1, \ldots, n$ and $b_j \in [\underline{b}_j, \overline{b}_j]$, $j = 1, \ldots, m$.

Let us denote by $u, s \in \mathbb{R}^N$ the input signal $\{u(t),\ t = 1, \ldots, N\}$ and the sequence of quantized measurements $\{s(t),\ t = 1, \ldots, N\}$, respectively. For a given input-output realization $\{u, s\}$ of length N, the feasible parameter set (FPS) is defined as:

$$\begin{aligned}\mathcal{F} = \bigcap_{t=1}^{N} \{\theta \in \Theta_0 :\ & \phi'(t)\,\theta \leq C_1 + \delta \quad \text{, if } s(t) = 0;\\ & C_1 - \delta < \phi'(t)\,\theta \leq C_2 + \delta \quad \text{, if } s(t) = 1;\\ & \vdots \\ & C_P - \delta < \phi'(t)\,\theta \quad \text{, if } s(t) = P\}.\end{aligned} \quad (4)$$

It is worth remarking that if the system outputs $y(t)$ were known, the FPS in (4) would be a convex polytope. If only the quantized measurements $s(t)$ are available, the FPS (4) is still a convex polytope if a FIR model is considered (i.e., $n = 0$). In fact, in such a case the regressor $\phi(t)$ is known. On the contrary, when an ARX model structure is considered, only quantized information on the autoregressive part of the regressor is available. Hence, the FPS turns out to be the projection on the parameter space of the augmented feasible set

$$\begin{aligned}\Omega = \bigcap_{t=1}^{N} \{[\theta'\ Y']' \in \Theta_0 \times \mathbb{R}^N :\ & \phi'(t)\,\theta \leq C_1 + \delta \text{ , if } s(t) = 0;\\ & C_1 - \delta < \phi'(t)\,\theta \leq C_2 + \delta \text{ , if } s(t) = 1;\\ & \vdots \\ & C_P - \delta < \phi'(t)\,\theta \text{ , if } s(t) = P\}.\end{aligned} \quad (5)$$

where $Y' = [y(1), \ldots, y(N)]$.

The set (5) is in general a nonconvex set in $\mathbb{R}^{n+m+N}$ and therefore also the FPS (4) turns out to be nonconvex, and possibly even disconnected.

III. Feasible set approximation

The aim of this paper is to construct an outer box approximation of the feasible set $\mathcal{F}$ for given N input/output pairs and a priori information on the model parameters.

Let us state the following technical assumptions which will be enforced throughout the paper.

Assumption 1: There exists $t_1 \neq t_2$ ($1 \leq t_1 \leq N$, $1 \leq t_2 \leq N$) such that $s(t_1) \neq s(t_2)$.

Assumption 2: The prior information on the parameters a_i is such that $a_i \in [\underline{a}_i, \overline{a}_i]$, where $\underline{a}_i > -\infty$ and $\overline{a}_i < \infty$, $i = 1, \ldots, n$. The a priori information on b_j is such that $b_j \in [\underline{b}_j, \overline{b}_j]$, where $\underline{b}_j \geq -\infty$ and $\overline{b}_j \leq \infty$, $j = 1, \ldots, m$.

Assumption 3: Thresholds $C_0 = -\infty$ and $C_{P+1} = \infty$ are replaced by $C_0 = \underline{M}$ and $C_{P+1} = \overline{M}$, where $-\infty < \underline{M} < \overline{M} < \infty$.

All these assumptions are very mild. Assumption 1 requires that the system output cannot be the same for all time points $t=1,\ldots,N$, while Assumption 2 states that an a priori knowledge on the bounds of parameters a_i must be available. Assumption 3 defines a bound on the output signal $y(t)$. From a practical point of view, thanks to prior knowledge on the system, it is not difficult to set these bounds properly.

For a measured output $s(t)$, let us denote by $\underline{y}(t)$ and $\overline{y}(t)$ the lower and upper bounds of the signal $y(t)$, respectively. In other words, if $s(t) = k$, according to (2), one has $\underline{y}(t) = C_k$ and $\overline{y}(t) = C_{k+1}$, and so $y(t) \in [\underline{y}(t), \overline{y}(t)]$.[1]

Assuming to have information on the system output from $t \geq 1$, it is easy to show that the feasible set $\mathcal{F}$ is the set of θ satisfying the following constraints:

$$\begin{cases} -\delta \leq y(t) - \sum_{i=1}^{n} a_i\, y(t-i) - \sum_{j=1}^{m} b_j\, u(t-j+1) \leq \delta \text{ , } t = r+1, \ldots, N \\ y(t) \in [\underline{y}(t), \overline{y}(t)] \text{ , } t = 1, \ldots, N \\ a_i \in [\underline{a}_i, \overline{a}_i] \text{ , } i = 1, \ldots, n \\ b_j \in [\underline{b}_j, \overline{b}_j] \text{ , } j = 1, \ldots, m \end{cases} \quad (6)$$

where $r = \max\{n, m-1\}$.

Since the first constraints of (6) are nonlinear in the variables a_i and $y(t-1)$, the resulting feasible set $\mathcal{F}$ is in general nonconvex (see Section VI for some examples of feasible sets).

Let us denote by $\mathcal{B}^* = \{\theta\colon a_i \in [\underline{a}_i^*, \overline{a}_i^*],\ b_j \in [\underline{b}_j^*, \overline{b}_j^*]\}$ the minimum outer box containing the set $\mathcal{F}$, i.e., for any set $\mathcal{B}^{(k)} = \{\theta\colon a_i \in [\underline{a}_i^{(k)}, \overline{a}_i^{(k)}],\ b_j \in [\underline{b}_j^{(k)}, \overline{b}_j^{(k)}]\}$ such that $\mathcal{B}^{(k)} \supseteq \mathcal{F}$, one has $\mathcal{B}^* = \bigcap_k \mathcal{B}^{(k)}$.

[1] With a slight abuse of notation we will always denote feasible sets by closed intervals.

Now, the aim is to find a box $\mathcal{B}$ which contains the optimal outer approximation $\mathcal{B}^*$, through a suitable convex relaxation of problem (6).

Let us define for $i = 1, \ldots, n$, $t = 1, \ldots, N$

$$\begin{aligned} w_i(t) &= (a_i - \underline{a}_i)(y(t) - \underline{y}(t)) \\ &= a_i y(t) - a_i \underline{y}(t) - \underline{a}_i y(t) + \underline{a}_i \underline{y}(t). \end{aligned} \tag{7}$$

Notice that, thanks to Assumption 2 and 3, $\underline{a}_i$ and $\underline{y}(t)$ are finite. By introducing the new variables $w_i(t)$, the set of inequalities (6) can be rewritten as

$$\begin{cases} -\delta \leq y(t) - \sum_{i=1}^{n} \big(w_i(t-1) + a_i \underline{y}(t-1) + \underline{a}_i y(t-1) - \underline{a}_i \underline{y}(t-1)\big) \\ \qquad - \sum_{j=1}^{m} b_j \, u(t-j+1) \leq \delta \ , t = r+1, \ldots, N \\ y(t) \in [\underline{y}(t), \overline{y}(t)] \ , t = 1, \ldots, N \\ a_i \in [\underline{a}_i, \overline{a}_i] \ , i = 1, \ldots, n \\ b_j \in [\underline{b}_j, \overline{b}_j] \ , j = 1, \ldots, m \\ w_i(t) = (a_i - \underline{a}_i)(y(t) - \underline{y}(t)) \ , i = 1, \ldots, n \ , t = 1, \ldots, N \end{cases} \tag{8}$$

where the nonlinear constraints are now embedded in the new variables $w_i(t)$.

For each $i = 1, \ldots, n,\ t = 1, \ldots, N$, let us relax the equality $w_i(t) = (a_i - \underline{a}_i)(y(t) - \underline{y}(t))$ to the following inequalities

$$0 \leq w_i(t) \leq (\overline{a}_i - \underline{a}_i)(y(t) - \underline{y}(t)) \tag{9}$$

$$0 \leq w_i(t) \leq (a_i - \underline{a}_i)(\overline{y}(t) - \underline{y}(t)). \tag{10}$$

By substituting (9)-(10) to the definition of $w_i(t)$ in (8), one obtains a set of linear constraints. Let us now introduce the following optimization problem which allows one to compute a lower bound on $\underline{a}_i^*$ (a similar problem can be formulated to compute an upper bound on $\overline{a}_i^*$).

$$\begin{cases} \inf \ \max \left\{ \max_{t=1,\ldots,N} \left\{ \dfrac{w_i(t)}{(y(t) - \underline{y}(t))} \right\}, (a_i - \underline{a}_i) \right\} \\ \text{s.t.:} \\ -\delta \leq y(t) - \sum_{i=1}^{n} \big(w_i(t-1) + a_i \underline{y}(t-1) + \underline{a}_i y(t-1) - \underline{a}_i \underline{y}(t-1)\big) \\ \qquad - \sum_{j=1}^{m} b_j \, u(t-j+1) \leq \delta \ , t = r+1, \ldots, N \\ y(t) \in [\underline{y}(t), \overline{y}(t)] \ , t = 1, \ldots, N \\ a_i \in [\underline{a}_i, \overline{a}_i] \ , i = 1, \ldots, n \\ b_j \in [\underline{b}_j, \overline{b}_j] \ , j = 1, \ldots, m \\ 0 \leq w_i(t) \leq (\overline{a}_i - \underline{a}_i)(y(t) - \underline{y}(t)) \ , i = 1, \ldots, n \ , t = 1, \ldots, N \\ 0 \leq w_i(t) \leq (a_i - \underline{a}_i)(\overline{y}(t) - \underline{y}(t)) \ , i = 1, \ldots, n \ , t = 1, \ldots, N \end{cases} \tag{11}$$

where the optimization variables are $y(t)$, $w_i(t)$, a_i, b_j, for $t = 1, \ldots, N$, $i = 1, \ldots, n$, $j = 1, \ldots, m$.

Problem (11) is a so-called *generalized linear fractional programming (GLFP)* problem [17, p. 152]. In fact, the objective function is the maximum of ratios of affine functions of the problem variables, while all the constraints are linear. Thus, (11) turns out to be a quasiconvex optimization problem. To solve this problem it is customary to use a bisection algorithm. For a given tolerance $\varepsilon > 0$, this method requires the solution of $\lceil \log_2((\overline{a}_i - \underline{a}_i)/\varepsilon) \rceil$ linear feasibility problems.

Let us state a proposition which follows directly from the previously discussed relaxation.

Proposition 1: For any $\theta \in \mathcal{F}$, there exist $w_i(t)$, $i = 1, \ldots, n$, $t = 1, \ldots, N$ such that problem (11) is feasible. Moreover, if $\hat{a}_i$ is the solution of problem (11), one has $\hat{a}_i \leq \underline{a}_i^*$.

In general, due to the relaxations of the equality constraint on $w_i(t)$, the solution of (11) provides a conservative lower bound on $\underline{a}_i^*$. This procedure must be repeated for all $i = 1, \ldots, n$. Since the parameter update is performed for one parameter at a time, in order to reduce conservatism it is convenient to iteratively repeat the entire procedure until convergence. To this purpose, a simple procedure written in pseudo-code is reported in Algorithm 1.

Algorithm 1 Compute bounds on a

1: Set tolerance ϵ
2: $\underline{q} \leftarrow \underline{a}$; $\overline{q} \leftarrow \overline{a} + 2\,\epsilon$;
3: **while** (norm($[\underline{q}; \overline{q}] - [\underline{a}; \overline{a}]$,'inf')$> \epsilon$) **do**
4: $\quad \underline{q} \leftarrow \underline{a}$; $\overline{q} \leftarrow \overline{a}$;
5: $\quad$ **for** $i \leftarrow 1 : n$ **do**
6: $\quad\quad \underline{a}_i \leftarrow$ LOWER_BOUND$(i, \underline{a}, \overline{a}, \underline{b}, \overline{b}, u, \delta, \underline{y}, \overline{y})$;
7: $\quad\quad \overline{a}_i \leftarrow$ UPPER_BOUND$(i, \underline{a}, \overline{a}, \underline{b}, \overline{b}, u, \delta, \underline{y}, \overline{y})$;
8: $\quad$ **end for**
9: **end while**

Let us define $\underline{a} = [\underline{a}_1, \ldots, \underline{a}_n]'$, $\overline{a} = [\overline{a}_1, \ldots, \overline{a}_n]'$, $\underline{b} = [\underline{b}_1, \ldots, \underline{b}_m]'$ and $\overline{b} = [\overline{b}_1, \ldots, \overline{b}_m]'$. The stopping condition in line 3 corresponds to

$$\left\| \begin{bmatrix} \underline{q} \\ \overline{q} \end{bmatrix} - \begin{bmatrix} \underline{a} \\ \overline{a} \end{bmatrix} \right\|_\infty \leq \epsilon \ . \tag{12}$$

So, if no parameter improves more than ϵ w.r.t. the previous iteration, the algorithm stops providing the final bounds $\underline{a}$, $\overline{a}$. In lines 5-8, a loop for each $i = 1, \ldots, n$ is performed for improving bounds on a_i. Here, function LOWER_BOUND (and a similar function UPPER_BOUND for updating the bound on $\overline{a}_i$) denotes a function able to solve problem (11) by standard bisection techniques.

Once Algorithm 1 is over, the final values of $\underline{a}_i$, $\overline{a}_i$ are available for all i. It remains to compute bounds on parameters b_j. This can be done by solving $2m$ LP problems. Such problems have the same constraints of (11), while the objective function is $\inf\, b_j$ to compute the lower bound on $\underline{b}_j^*$ (and $\sup\, b_j$ for the upper bound on $\overline{b}_j^*$), for $j = 1, \ldots, m$. In general, the obtained bounds are not tight due to conservatism of the proposed relaxation of the FPS constraints.

IV. Improving convergence of the bounding procedure

In the previous section, an algorithm has been proposed to iteratively refine the bounds on the parameters a_i of the ARX model. Unfortunately, Algorithm 1 may execute several iterations before the stopping condition (12) is satisfied, thus requiring the solution of a large number of LPs.

In this section, a new algorithm is proposed in order to improve the convergence rate of the bounding procedure reducing the number of LPs to be solved (see the numerical tests presented in Section VI). Moreover, this alternative procedure is proven to provide tight bounds in the case of first-order ARX models.

The new algorithm is based on the solution of a sequence of LPs of the type

$$\begin{cases} \inf \ a_i \\ \text{s.t.:} \\ -\delta \leq y(t) - \sum_{i=1}^{n} \big(w_i(t-1) + a_i \underline{y}(t-1) + \underline{a}_i y(t-1) - \underline{a}_i \underline{y}(t-1) \big) \\ \qquad - \sum_{j=1}^{m} b_j \, u(t-j+1) \leq \delta \ , t = r+1, \ldots, N \\ y(t) \in [\underline{y}(t), \overline{y}(t)] \ , t = 1, \ldots, N \\ a_i \in [\underline{a}_i, \overline{a}_i] \ , i = 1, \ldots, n \\ b_j \in [\underline{b}_j, \overline{b}_j] \ , j = 1, \ldots, m \\ 0 \leq w_i(t) \leq (\overline{a}_i - \underline{a}_i)(y(t) - \underline{y}(t)) \ , i = 1, \ldots, n \, , t = 1, \ldots, N \\ 0 \leq w_i(t) \leq (a_i - \underline{a}_i)(\overline{y}(t) - \underline{y}(t)) \ , i = 1, \ldots, n \, , t = 1, \ldots, N. \end{cases} \tag{13}$$

Notice that (13) has the same constraints of (11), while its solution provides a lower bound to the solution $\hat{a}_i$ of problem (11). Hence Proposition 1 holds also for (13). Proposition 1 provides necessary conditions for the feasibility of a given θ, i.e., if (13) is not feasible for a given θ, then $\theta \notin \mathcal{F}$. This is a key information which will be exploited to speed up convergence of the bounding procedure. For ease of presentation, let us define the following function returning a solution of (13):

$$[feas, opt] = \texttt{solve\_lp}(obj, \underline{a}, \overline{a}, \underline{b}, \overline{b}, u, \delta, \underline{y}, \overline{y})$$

where obj denotes the objective function to be minimized or maximized (e.g., $obj =$ '$\inf a_i$' in (13)), $feas$ can be true or false depending on the feasibility of the problem, while opt contains the optimal solution (in case of feasible problems).

Algorithm 2 describes a procedure which returns a lower bound on the regression parameter a_i (a similar procedure can be devised for the upper bound).

Algorithm 2 Improved lower bound on parameter a_i

1: **function** LOWER_BOUND_IMPR($i, \varepsilon, \underline{a}, \overline{a}, \underline{b}, \overline{b}, u, \delta, \underline{y}, \overline{y}$)
2: $\quad \underline{q} \leftarrow \underline{a}$; $\overline{q} \leftarrow \overline{a}$; $step \leftarrow (\overline{q}_i - \underline{q}_i)/2$;
3: $\quad$ **while** ($step \geq \varepsilon/2$) **do**
4: $\qquad \overline{q}_i \leftarrow \underline{q}_i + step$;
5: $\qquad [feas, opt]$=solve_LP('$\inf a_i$',$\underline{q}, \overline{q}, \underline{b}, \overline{b}, u, \delta, \underline{y}, \overline{y}$)
6: $\qquad$ **if** $feas$ **then**
7: $\qquad\quad step \leftarrow step/2$; $\underline{q}_i \leftarrow opt$;
8: $\qquad$ **else**
9: $\qquad\quad \underline{q}_i \leftarrow \overline{q}_i$;
10: $\qquad$ **end if**
11: $\quad$ **end while**
12: $\quad \underline{a}_i \leftarrow \underline{q}_i$;
13: $\quad$ **return** $\underline{a}$
14: **end function**

The scalar $step$ denotes the size of the interval on parameter a_i which will be analyzed at the current iteration. Throughout Algorithm 2, $\underline{q} = \underline{a}$, $\overline{q} = \overline{a}$, except for the i-th component $\underline{q}_i$ and $\overline{q}_i$ which denote the current estimated bounds on the lower bound $\underline{a}_i$, i.e., $\underline{a}_i \in [\underline{q}_i, \overline{q}_i]$ at the current iteration. In line 5, the LP (13) is solved in the interval $[\underline{q}, \overline{q}]$. If it is feasible, the lower bound $\underline{q}_i$ is updated with the solution of the LP, while the value of $step$ is divided by two. If it is infeasible, $\underline{q}_i$ is set to $\overline{q}_i$.

In order to analyze the convergence properties of Algorithm 2, let us first introduce the following assumption.

Assumption 4: There exists a constant $\varepsilon > 0$ such that, if $\overline{a}_i - \underline{a}_i < \varepsilon$ and $[\underline{a}_i, \overline{a}_i] \cap [\underline{a}_i^*, \overline{a}_i^*] = \emptyset$, then the set of constraints in the LP (13) is infeasible.

Note that Assumption 4 is not restrictive, because when the constant ε approaches zero, by (9)-(10), also $w_i(t) = 0$, $\forall t$, and hence the relaxed constraint set (13) boils down to the exact one (8) (said another way, infeasibility of a given a_i can be checked exactly by solving a single LP).

Theorem 1: Algorithm 2 converges in a finite number of iterations. Moreover, if $n = 1$ and Assumption 4 holds, then the lower bound $\underline{a}_1$ returned by Algorithm 2 is tight, i.e., $\underline{a}_1^* - \underline{a}_1 \leq \varepsilon$.

Proof: In Algorithm 2, every time the LP solved in line 5 is feasible, the variable $step$ is halved. Hence, being the stopping condition $step < \varepsilon/2$, in order to prove that such a condition is eventually satisfied, it is sufficient to show that infeasibility of the LP cannot occur indefinitely. This directly follows by the fact that the lower bound $\underline{q}_i$ is increased by $step$ every time infeasibility occurs (see lines 9 and 4). Now, let $n = 1$ and assume that the tolerance ε in Algorithm 2 is chosen according to Assumption 4. Whenever $\underline{a}_1^* - \underline{q}_1 > \varepsilon$, in Algorithm 2 either $\underline{q}_1$ is increased by $step > \varepsilon$, or the variable $step$ is halved. This will eventually lead either to a value of $\underline{q}_1$ satisfying $\underline{a}_1^* - \underline{q}_1 \leq \varepsilon$, or to $step \leq \varepsilon$. In the latter case, Assumption 4 guarantees that all the subsequent LPs with the constraint $a_1 \in [\underline{q}_1, \underline{q}_1 + step]$ will be infeasible as long as $\underline{q}_1 + step < \underline{a}_1^*$. Therefore, feasibility will finally occur only when $\underline{a}_1^* - \underline{q}_1 \leq step \leq \varepsilon$, and then $step$ is halved for the last time and the stopping condition is satisfied. ■

Theorem 1 states that for first-order ARX models one can find the exact bounds on the parameter a_1 by executing a single run of Algorithm 2. Although this property does not hold for general ARX models, it is expected that using Algorithm 2 to bound each parameter a_i may reduce the number of iterations needed to achieve convergence. Therefore, for a generic ARX of order n it is proposed to use Algorithm 1, with the function LOWER_BOUND replaced by the function LOWER_BOUND_IMPR described by Algorithm 2 (and similarly for the function UPPER_BOUND). In Section VI, it will be shown by numerical tests that this allows to significantly reduce the overall number of LPs to be solved in the bounding procedure.

V. Structural properties of the FPS for binary sensors

In this section, we analyze some structural properties of the FPS, for the case of a binary sensor

$$s(t) = \sigma(y(t)) \triangleq \begin{cases} 0 & \text{if } y(t) \leq C \\ 1 & \text{if } y(t) > C \end{cases} \tag{14}$$

Theorem 2: For any output sequence $s \in \{0,1\}^N$, any input sequence $u \in \mathbb{R}^N$ and any noise level δ, the set

$$\{\theta \in \Theta_0 : \ b_1 = b_2 = \cdots = b_m = 0; \ \ \textstyle\sum_{i=1}^{n} a_i = 1\} \tag{15}$$

is always contained in the feasible set $\mathcal{F}$.

Proof: Let $\epsilon > 0$ and set

$$y(t) = \begin{cases} C + \epsilon & \text{if } s(t) = 1 \\ C - \epsilon & \text{if } s(t) = 0 \end{cases} . \tag{16}$$

Then, for every t, the second constraint in (6) is satisfied by construction, while the first constraint boils down to

$$-\delta \leq C \pm \epsilon - \sum_{i=1}^{n} a_i \,(C \pm \epsilon) - \sum_{j=1}^{m} b_j \, u(t-j+1) \leq \delta. \quad (17)$$

By setting $b_1 = \ldots = b_m = 0$ and $a_1 + a_2 + \cdots + a_n = 1$, (17) becomes $-\delta \leq \pm\epsilon - \sum_{i=1}^{n} \pm a_i \, \epsilon \leq \delta$ which is satisfied for any $\delta > 0$, provided that a sufficiently small ϵ is chosen. ■

Remark 1: Theorem 2 shows that, in the case of binary sensors, the hyperplane defined by the equality constraints in (15) is always contained in the FPS, unless this is excluded by suitable a priori information on the system dynamics (i.e., by choosing Θ_0 so that the set (15) is empty). Notice that the constraint $\sum_{i=1}^{n} a_i = 1$ implies that the system transfer function has a pole in 1, which corresponds to the fact that the system can "hold" indefinitely the constant value C even with no input signal, thus letting an arbitrarily small noise generate any binary output sequence s.

Now, let us consider the special case of an ARX(1,1) model $y(t) = a_1 y(t-1) + b_1 u(t) + d(t)$. The following result holds.

Corollary 1: Let $u, s \in \mathbb{R}^N$ be given and $\underline{u} = \min_t u(t)$, $\overline{u} = \max_t u(t)$. Define the set

$$\mathcal{I} = \Big\{\theta \in \Theta_0 : \quad 1 - \frac{\delta}{C} \leq a_1 + \frac{\underline{u}}{C} b_1 \leq 1 + \frac{\delta}{C} \qquad 1 - \frac{\delta}{C} \leq a_1 + \frac{\overline{u}}{C} b_1 \leq 1 + \frac{\delta}{C} \Big\}. \quad (18)$$

Then, $\mathcal{I} \subseteq \mathcal{F}$.

Proof: As in the proof of Theorem 2, choose $y(t)$ as in (16). The first constraint in (6) becomes

$$-\delta \leq C \pm \epsilon - a_1(C \pm \epsilon) - b_1 u(t) \leq \delta$$

which is equivalent to

$$1 - \frac{\delta}{C} \leq a_1 + \frac{u(t)}{C} b_1 + g(\epsilon) \leq 1 + \frac{\delta}{C} \quad (19)$$

where $g(\epsilon) = \pm a_1 \frac{\epsilon}{C} \mp \frac{\epsilon}{C}$. For $\epsilon \to 0$, the inequalities (19) represent a strip in the (a_1, b_1)-plane. The intersections of such strips for $t = 1, \ldots, N$ provides the set $\mathcal{I}$ in (18). ■

Corollary 1 highlights that, for a given data set, there is always a set of nonzero measure contained in the FPS (namely, a parallelogram), which does not depend on the output provided by the binary sensor, but only on the extreme values taken by the input signal. Notice that for arbitrarily large input signals and arbitrarily small δ, the set $\mathcal{I}$ boils down to the point $\{\theta : \; b_1 = 0, \; a_1 = 1\}$, in accordance with Theorem 2. The result in Corollary 1 can be easily generalized to ARX models of arbitrary order.

VI. Examples

In this section, three examples are reported to show the effectiveness of the proposed work.

Example 1: Consider the following ARX system of order 1

$$y(t) = 0.6\, y(t-1) + 0.15\, u(t) + d(t) \; .$$

Let us assume that the prior knowledge on the system is $a_1 \in [-10; 10]$ and $|d| \leq 0.1$. The system output $y(t)$ is measured by a binary sensor with threshold $C = 1$. An identification experiment of length $N = 200$ is performed by choosing an input signal $u(t)$ uniformly distributed in [-10;10]. The noise $d(t)$ has been generated with a uniform distribution in [-0.1;0.1]. A priori bounds on the output are taken as $\underline{M} = -10$ and $\overline{M} = 10$ (see Assumption 3). In Fig. 1, the unknown signal $y(t)$ and its corresponding binary sequence $s(t)$ are reported. By applying the procedure proposed in Section IV with tolerance $\varepsilon = 10^{-6}$, one obtains the outer box $a_1 \in [0.49884; 1.10000]$, $b_1 \in [-0.01037; 0.23891]$, while the minimum outer box containing the feasible set is $a_1 \in [0.49884; 1.10000]$, $b_1 \in [-0.01033; 0.20099]$. The true feasible set and the computed approximating box are reported in Fig. 2. In order to draw the feasible set, a gridding technique has been used. As stated by Theorem 1, the bounds on parameter a_1 are tight. The total number of solved LPs is 67 for a computation time of about 1.2 seconds. The number of solved LPs by using the procedure described in Section III turns out to be 649 for a computation time of about 12 seconds.[2]

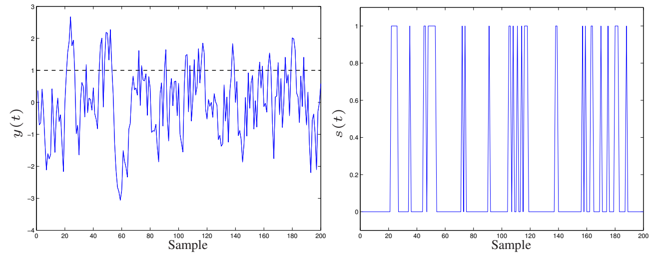

Fig. 1. Example 1. Unknown output sequence $y(t)$ and corresponding binary measurements $s(t)$, $t = 1, \ldots, 200$.

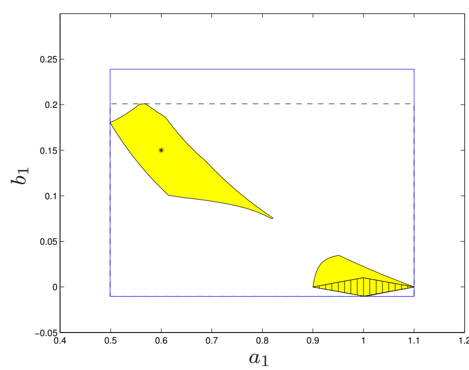

Fig. 2. Example 1. Feasible set, computed outer box (solid) and minimum outer box (dashed). The true parameter vector is marked by '*'. The striped region corresponds to the set $\mathcal{I}$ in (18).

Example 2: Consider the following ARX system

$$y(t) = 1.55y(t-1) - 0.86y(t-2) + 3.25u(t) + 1.83u(t-1) + d(t).$$

Let quantized measurements be given in the range $[-30; 30]$ with a resolution of 1. Prior information on the system are $a_1, a_2 \in [-100; 100]$ and $|d| \leq 1$. An identification experiment of length 100 is performed by applying a uniformly distributed input in $[-1; 1]$. In Table I, a comparison between tight bounds (computed by gridding) and computed bounds are reported. The total number of solved LPs is 236

[2] Computations have been performed under Matlab by using *CPLEX* [18], [19] to solve the LPs, on an Intel Core i5 M520 at 2.40 GHz with 4 GB of RAM.

TABLE I

EXAMPLE 2. TIGHT AND COMPUTED BOUNDS FOR THE FEASIBLE SET.

	tight bounds		computed bounds	
	min	max	min	max
a_1	1.457	1.649	1.450	1.650
a_2	-0.966	-0.773	-0.969	-0.772
b_1	1.943	4.634	1.855	4.683
b_2	0.326	2.689	0.298	2.847

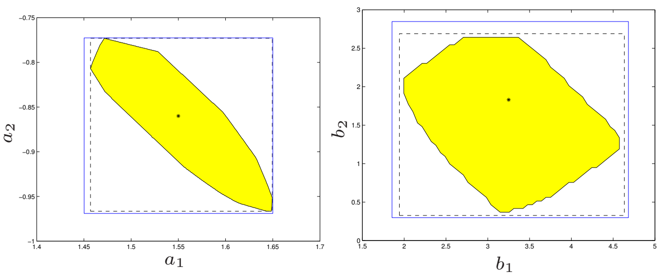

Fig. 3. Example 2. Projection of the feasible set on the plane (a_1, a_2) [left] and (b_1, b_2) [right]. Computed outer box (solid) and minimum outer box (dashed). The true parameter vector is marked by '*'.

for a computation time of about 2.8 seconds. In Figure 3, the projection of the true feasible set on the plane (a_1, a_2) and (b_1, b_2) are reported along with the computed bounds. Notice that, in order to make the computational burden acceptable, it has been necessary to coarsen significantly the grid (especially w.r.t. parameters b_j).

Example 3: Consider an ARX system with $n = m = 4$. Let quantized measurements be given in the range $[-350; 350]$. Prior information on the system are $a_i \in [-100; 100]$, $i = 1, \ldots, 4$ and $|d| \leq 1$. An identification experiment of length $N = 300$ is performed by applying a uniformly distributed input in $[-5; 5]$.

In Table II, the true parameter values and the computed outer boxes are reported assuming a sensor resolution of 0.5 and 5, respectively. For both cases, the computation time is less than 6 minutes. The true bounds cannot be computed in this case because for an 8-dimension parameter space the gridding approach is not computationally tractable.

VII. CONCLUSIONS

In this paper, an algorithm for bounding the feasible parameter set of ARX models in presence of binary or quantized measurements has been proposed. The measurement noise is assumed to be unknown but bounded and a worst-case approach has been adopted. Convergence issues have been analyzed and the proposed algorithm turns out to be computationally feasible for ARX systems of reasonable order. Another result concerns the characterization of some structural properties of the feasible parameter set when measurements are obtained through binary sensors. The investigation of similar properties in the case of multi-threshold sensors is the subject of ongoing research.

TABLE II

EXAMPLE 3. TRUE PARAMETER VALUES AND COMPUTED BOUNDS FOR THE FEASIBLE SET.

	true values	computed bounds (sensor resolution 0.5)		computed bounds (sensor resolution 5)	
		min	max	min	max
a_1	1.33	1.291	1.364	1.105	1.552
a_2	-0.66	-0.724	-0.598	-1.026	-0.361
a_3	0.58	0.527	0.649	0.256	0.923
a_4	-0.41	-0.443	-0.385	-0.572	-0.256
b_1	8.25	8.108	8.410	7.180	9.190
b_2	3.83	3.556	4.153	1.989	5.641
b_3	5.15	4.889	5.511	3.029	7.461
b_4	-3.12	-3.326	-2.901	-4.148	-1.658

REFERENCES

[1] L. Y. Wang, J. F. Zhang, and G. G. Yin. System identification using binary sensors. *IEEE Transactions on Automatic Control*, 48(11):1892–1907, 2003.

[2] J. Juillard, K. Jafaridinani, and E. Colinet. Consistency of weighted least-square estimators for parameter estimation problems based on binary measurements. In *15th IFAC Symposium on System Identification*, pages 72–77, Saint-Malo, France, July 2009.

[3] Y. Zhao, L. Y. Wang, G. Yin, and J. Zhang. Identification of wiener systems with binary-valued output observations. *Automatica*, 43(10):1752 – 1765, 2007.

[4] G. Yin, L. Y. Wang, and S. Kan. Tracking and identification of regime-switching systems using binary sensors. *Automatica*, 45(4):944 – 955, 2009.

[5] J. C. Aguero, B. I. Godoy, G. C. Goodwin, and T. Wigren. Scenario-based EM identification for FIR systems having quantized output data. In *15th IFAC Symposium on System Identification*, pages 66–71, Saint-Malo, France, July 2009.

[6] L.Y. Wang, C. Li, G. Yin, L. Guo, and C. Xu. State observers and output feedback of lti systems with binary observations. In *15th IFAC Symposium on System Identification*, pages 84–89, Saint-Malo, France, July 2009.

[7] L. Y. Wang, G. G. Yin, J. F. Zhang, and Y. Zhao. *System Identification with Quantized Observations*. Springer, 2010.

[8] F. Gustafsson and R. Karlsson. Estimation based on quantized information. In *15th IFAC Symposium on System Identification*, pages 78–83, Saint-Malo, France, July 2009.

[9] M. Milanese and A. Vicino. Optimal estimation theory for dynamic systems with set membership uncertainty: an overview. *Automatica*, 27(6):997–1009, 1991.

[10] A. Garulli, A. Tesi, and A. Vicino, editors. *Robustness in Identification and Control*. Lecture Notes in Control and Information Sciences. Springer, London, 1999.

[11] M. Casini, A. Garulli, and A. Vicino. Time complexity and input design in worst-case identification using binary sensors. In *Proc. 46th IEEE Conference on Decision and Control*, pages 5528–5533, New Orleans, USA, December 2007.

[12] M. Casini, A. Garulli, and A. Vicino. Optimal input design for identification of systems with quantized measurements. In *Proc. 47th IEEE Conf. on Decision and Control*, pages 5506–5512, Cancun, Mexico, December 2008.

[13] M. Casini, A. Garulli, and A. Vicino. Input design for worst-case system identification with uniformly quantized measurements. In *15th IFAC Symposium on System Identification*, pages 54–59, Saint-Malo, France, July 2009.

[14] Y. Zhao, L.Y. Wang, J. Zhang, and G. Yin. Jointly deterministic and stochastic identification of linear systems using binary-valued observations. In *15th IFAC Symposium on System Identification*, pages 60–65, Saint-Malo, France, July 2009.

[15] V. Cerone, D. Piga, and D. Regruto. Set-membership EIV identification through LMI relaxation techniques. In *Proc. IEEE American Control Conference*, pages 2158–2163, Baltimore, USA, July 2010.

[16] V. Cerone, D. Piga, and D. Regruto. Improved parameters bounds for set-membership EIV problems. *International Journal of Adaptive Control and Signal Processing*, 25(3):208–227, 2011.

[17] S. Boyd and L. Vandenberghe. *Convex Optimization*. Cambridge University Press, Cambridge (UK), 2004.

[18] IBM. IBM ILOG Cplex Optimizer. Online: http://www-01.ibm.com/software/integration/optimization/cplex-optimizer/.

[19] M. Baotic. Cplexint - Matlab interface for the Cplex solver. Online: http://control.ee.ethz.ch/∼hybrid/cplexint.php.