POLITECNICO
MILANO 1863

SCUOLA DI INGEGNERIA INDUSTRIALE
E DELL'INFORMAZIONE

EXECUTIVE SUMMARY OF THE THESIS

State Estimation with Switching Measurements: Sensor Scheduling and Existence of Observable Schedules

TESI MAGISTRALE IN AUTOMATION AND CONTROL ENGINEERING – INGEGNERIA DELL'AUTOMAZIONE

AUTHOR: AMIN BIGLARY MAKVAND

ADVISOR: PROF. ALESSANDRO COLOMBO

CO-ADVISOR: PROF. MARCELLO FARINA

ACADEMIC YEAR: 2021-2022

1. Introduction

With the emergence of large sensor networks, there is a growing need for algorithms that can decide, for specific objectives, which sensors in a large network of sensors should be used at each time step. Regarding linear time-invariant systems that are under the measurement of large sensor networks, such sensors should grant observability and, at the same time, should minimize the state estimation error covariance. In the literature, most works are concerned with the minimization of the error covariance. This work aims to establish the foundation necessary to investigate the observability of such systems. When different sensors are selected at different time steps, the system measurement vector is time-variant. Thus, we are dealing with systems under time-variant measurement schemes. Therefore, we first attempt to extend the concept of observability to the case of linear systems with time-variant measurement. Then, we investigate conditions that allow the existence of observable sensor schedules first presented in [1]. We mainly focus on a theorem on the existence of observable schedules, and we try to restructure and simplify it by introducing new definitions and lemmas. As a result, an algorithm is introduced. This algorithm can find a subset of sensors in a set of available sensors that can be used to construct an observable sensor schedule. The algorithm is then numerically implemented. The results of the numerical implementation of the algorithm support the claim of the theorem and provide further insights on the structure of the observable sensor schedule.

2. Observability for linear discrete-time systems with time-variant measurements

Observability is a condition that determines if it is possible to infer the state of a system from the knowledge of system outputs over a finite period. This concept was introduced by [2, 3].

Consider the following state equation, where $C \in \mathbb{R}^{n_c \times n}$,

$$x(k+1) = Ax(k), \quad (2.1)$$
$$y(k) = C_k x(k).$$

From now on, we denote by k_0 the initial time instant.

Definition 2.1. (*Measurement Time Horizon*). The Measurement time horizon T_N is defined as a sequence of time steps $\{k_0, k_0+1, \dots, k_0+N-1\}$ in which the system is under measurement. ■

Definition 2.2. (*N-horizon Observability*). Given the state equation (2.1), a state $x_0 \in \mathbb{R}^n$ is unobservable over a measurement time horizon $T_N = \{k_0, k_0+1, \dots, k_0+N-1\}$ if setting $x(k_0) = x_0$ as the initial state, $y(k) \equiv 0$ for all $k \in T_N$. The state equation is N-horizon observable if the zero vector $0 \in \mathbb{R}^n$ is the only unobservable state. ■

Theorem 2.1. The proposed linear system (2.1) is N-horizon observable over the measurement time horizon $T_N = \{k_0, k_0+1, \dots, k_0+N-1\}$ if and only if $\text{rank}(\phi_N) = n$, where

$$\phi_N = \begin{bmatrix} C_{k_0} \\ C_{k_0+1}A \\ C_{k_0+2}A^2 \\ \cdots \\ C_{k_0+N-1}A^{N-1} \end{bmatrix}.$$

3. Existence of an observable schedule

In this section, we first define the concept of sensor schedule and its observability properties. Then, we explore the conditions that allow for the existence of an observable sensor schedule for a discrete-time linear system with time variant measurements. An algorithm drawn from [1] is proposed. If specific conditions discussed below are met, this algorithm can be used to construct an observable sensor schedule. A theorem introduced in [1] on the existence of observable schedules is restructured here and used as the basis of the proposed algorithm.

3.1. Problem formulation

The problem is formulated borrowing terms and notations from [4]. Consider the following linear system:

$$x(k+1) = Ax(k) \quad (3.1)$$

where $x(k) \in \mathbb{R}^n$ is the state of the system. Consider the ordered set $S = \{S[0], S[1], \dots, S[p-1]\}$, where $S[i] \in \mathbb{R}^n$ is a sensor vector that can be used to measure the system state. Each element of the set S represents one sensor.

Definition 3.1 (*Sensor Schedule*). A sensor schedule Σ over an ordered sensor set S is defined as a set of elements $\sigma_k \in \{0, \dots, p-1\}$ where $\sigma_k = i$ denotes the usage of sensor $S[\sigma_k] \in S$ at time k. ■

Definition 3.2 (*N-horizon Sensor Schedule*). Let T_N be the measurement time horizon (see Definition 2.3). Denote by $\Sigma_N = \{\sigma_{k_0}, \sigma_{k_0+1}, \dots, \sigma_{k_0+N-1}\}$, $\sigma_k \in \{0, \dots, p-1\}$ a N-horizon sensor schedule. ■

To measure the system state, only one sensor is allowed to operate at each time step. Under a given schedule Σ_N, the scheduled measurement at each time step is

$$y(k) = S[\sigma_k]^T x(k),$$
$$\forall\, k \in \{k_0, k_0+1, \dots, k_0+N-1\}.$$

3.2. Observable sensor schedule

Definition 3.3 (*Observable N-horizon Sensor Schedule*). A N-horizon sensor schedule with an observability matrix ϕ_N of rank n is called an observable N-horizon sensor schedule (see Theorem 2.1). ■

The following definitions are required to define conditions for N-horizon observability and the definition of observable N-horizon sensor schedules.

Definition 3.4 (*Cover Set*). Let $\alpha = [c_0 \dots c_{n-1}]^T, c_i \in \mathbb{R}$ be the coordinate vector of vector $v \in \mathbb{R}^n$, with respect to a basis $G = \{g_0, \dots, g_{n-1}\}$ spanning vector space $\mathbb{R}^n$. The cover set of vector v with respect to the basis G, denoted by β, is the set of all basis elements $g_i \in G$ with an associated $c_i \neq 0$. ■

Definition 3.5 (*Basis Coverage*)**.** A vector set $V = \{v_0, v_1, \dots, v_{n_v-1}\}, v_i \in \mathbb{R}^{n_v}$ covers the basis G when

$$g_i \in \cup_{l=0}^{n_v-1} \beta_l, \forall\, g_i \in G,$$

where each β_l is the cover set of its corresponding vector v_l, with respect to the basis G. ■

Definition 3.6 (*Non-common Element*)**.** A set β_a has a non-common element with respect to a set β_b, if there exists a $g_i \in G$ such that $g_i \in \beta_a$ and $g_i \notin \beta_b$. g_i is called a non-common element of β_a with respect to β_b. ■

3.3. Algorithm and theorem

What follows next is an algorithm that can be used to find a specific subset of S, denoted by S_o. Later in Theorem 3.1, it is indicated that elements of S_o under certain conditions, can be used to construct an observable sensor schedule. To proceed, consider the following assumption.

Assumption 3.1. Matrix A is nonsingular with distinct eigenvalues. ■

Consistently with Assumption 3.1, matrix A has n linearly independent eigenvectors. These eigenvectors are used as the elements of basis $G = \{g_0, \dots, g_{n-1}\}$. More specifically vectors $g_i \in \mathbb{R}^n$ are the left eigenvectors of A.
As mentioned, the proposed algorithm accepts ordered set S and returns the subset S_o. Here we repeat a short explanation of its main rational. The algorithm initializes a set S_r with S (line 1), then at each iteration, it selects a sensor vector from the set S_r, removes it from S_r (line 28) and adds it to S_o (line 24). To do so, the algorithm looks at the cover sets of every sensor vector that has in S_r (15 to 22). And selects the sensor vector (denoted B_o^0) with the largest set (denoted B_o^1) of non-common elements (see Definition 3.6) with respect to the union of previously chosen sets of non-common elements (lines 15 to 23). The algorithm then adds the largest set of the non-common elements B_o^1 to the set $B_\mathcal{M}$ (line 26), removes it from γ_r (line 27) and adds its corresponding sensor vector B_o^0 to S_o. The algorithm continues this process until one of two conditions is met: either there are no sensors left to check, corresponding to $S_r = \emptyset$, or there are no non-common elements left, corresponding to $\gamma_r = \emptyset$. Note that the elements in the cover sets are the vectors of basis G.

Algorithm 3.1 (*Sensor Set Observability Filter*)

1:	**ordered set** $S_r \leftarrow S$
2:	**ordered set** $\gamma_r \leftarrow G$
3:	**ordered set** $S_o \leftarrow \emptyset$
4:	**ordered set** $N_o \leftarrow \emptyset$
5:	**ordered set** $z \leftarrow \emptyset$
6:	**ordered set** $B_\mathcal{M} \leftarrow \emptyset$
7:	**ordered set** $\beta_{sub} \leftarrow \emptyset$
8:	**ordered set** $B_p \leftarrow \emptyset$
9:	**ordered set** $B_o^0 \leftarrow \emptyset$
10:	**ordered set** $B_o^1 \leftarrow \emptyset$
11:	**integer** $l \leftarrow 0$
12:	**integer** $\zeta \leftarrow 0$
13:	**while** $\gamma_r \neq \emptyset$ **or** $S_r \neq \emptyset$
14:	$l \leftarrow 0$
15:	**while** $l <$ **n**(S_r)
16:	**if** $B_\mathcal{M} = \emptyset$
17:	$\beta_{sub} \leftarrow$ **Cover**$(S_r[l], G)$
18:	$B_p \leftarrow$ **Append**$(B_p, (\beta_{sub}, S_r[l]))$
19:	**else**
20:	$\beta_{sub} \leftarrow$ **Cover**$(S_r[l], G)/\left(\left(\cup_{k=0}^{\mathbf{n}(B_\mathcal{M})-1} B_\mathcal{M}[k]\right) \cap$ **Cover**$(S_r[l], G)\right)$
21:	$B_p \leftarrow$ **Append**$(B_p, (\beta_{sub}, S_r[l]))$
22:	$l \leftarrow l + 1$
23:	$(B_o^0, B_o^1) \leftarrow$ **Max**(B_p)
24:	$S_o \leftarrow$ **Append**(S_o, B_o^1)
25:	$N_o \leftarrow$ **Append**$(N_o,$ **n**$(B_o^0))$
26:	$B_\mathcal{M} \leftarrow$ **Append**$(B_\mathcal{M}, B_o^0)$
27:	$\gamma_r \leftarrow \gamma_r / B_o^0$
28:	$S_r \leftarrow S_r / B_o^1$
29:	$\zeta \leftarrow \zeta +$ **n**(B_o^0)
30:	$z \leftarrow$ **Append**(z, ζ)
31:	$B_p \leftarrow \emptyset$
32:	**return** $S_o, N_o, B_\mathcal{M}, \gamma_r, z$

Assumption 3.2. Integer θ is equal to 1 + the number of elements in returned sets $S_o, N_o, B_\mathcal{M}$ and z when the algorithm terminates under the condition $\gamma_r = \emptyset$. ■

The following theorem can be proved, providing a fundamental result for defining observable sensor schedules.

Theorem 3.1. If S covers basis G (see Definition 3.5), then there exists an observable n-horizon sensor schedule $\Sigma_n = \{\sigma_{k_0}, \sigma_{k_0+1}, \dots, \sigma_{k_0+n-1}\}$ over sensor set S_o (see Definition 3.2), with the following structure:

$$\sigma_{k_0} = 0,$$
$$\vdots$$
$$\sigma_{k_0+z[0]-1} = 0,$$
$$\sigma_{k_0+z[0]} = 1,$$
$$\vdots$$
$$\sigma_{k_0+z[1]-1} = 1,$$

$$\vdots$$
$$\sigma_{k_0+z[\theta-1]} = \theta,$$
$$\vdots$$
$$\sigma_{k_0+z[\theta]-1} = \theta,$$

Such that $z[\theta] = n$. ∎

In this schedule, every sensor vector in the returned ordered set S_o is used. Each sensor is used $z[i] - z[i-1]$ number of times. Note that z is an ordered set with integer elements where each element $z[m] = \sum_{i=0}^{m} n(B_{\mathcal{M}}[i])$. Consider $z[-1] = 0$.

4. Algorithm numerical implementation

In this section, the proposed algorithm is implemented, and numerical examples are provided to verify the theoretical results of the previous section. The algorithm is implemented in Python language.

The code accepts the system matrix A, left eigenvectors of A, and the desired set of sensor vectors and returns several values. A Boolean value indicating whether the sensor set covers the basis; a list of sensor vectors corresponding to S_o; a list of integers N_o indicating the number of times the sensors of S_o should be repeated in the proposed schedule of Theorem 3.1 and finally, the rank of the observability matrix of the schedule of Theorem 3.1.

4.1. Examples

Let A be an arbitrary matrix that follows Assumption 3.1. Thus, it has n linearly independent eigenvectors,

$$A = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 2 & 11 & 0 & 0 & 0 \\ 4 & 4 & -8 & 0 & 0 \\ 5 & 8 & 0 & 7 & 0 \\ 7 & 7 & 3 & 0 & 6 \end{bmatrix}.$$

Consider basis G with left eigenvectors of A as its elements.

$$G = \{[1\ 0\ 0\ 0\ 0]^T, [\tfrac{-68}{171}\ \tfrac{-4}{19}\ 1\ 0\ 0]^T, [\tfrac{33}{35}\ \tfrac{-11}{7}\ \tfrac{3}{14}\ 0\ 1]^T, [\tfrac{1}{6}\ -2\ 0\ 1\ 0]^T, [\tfrac{1}{5}\ 1\ 0\ 0\ 6]^T\}.$$

Now consider the following examples. In each example, sensor set S is intentionally designed to cover different scenarios.

Example 4.1. Consider the following sensor vector set S, where S covers the basis G.

$$S = \{S[0] = [3.0332, -5.1353, 2.6428, 0, 3]^T,$$
$$S[1] = [5.1047, -22.2857, 0.8571, 8, 4]^T,$$
$$S[2] = [0.6, 3, 0, 0, 0]^T,$$
$$S[3] = [1.1666, -14, 0, 7, 0]^T,$$
$$S[4] = [1.2, 6, 0, 0, 0]^T$$
$$S[5] = [2.5784, -6.9172, 3.8571, 0, 4]\}$$

In the sensor set S, no sensor alone covers the basis G. Table 4.1 shows that no n-horizon observable sensor schedules can be constructed with only one sensor.

	σ_0	σ_1	σ_2	σ_3	σ_4	$\text{rank}(\phi_n)$
$\Sigma_{n,1}$	0	0	0	0	0	3
$\Sigma_{n,2}$	1	1	1	1	1	2
$\Sigma_{n,3}$	2	2	2	2	2	1
$\Sigma_{n,4}$	3	3	3	3	3	1
$\Sigma_{n,5}$	4	4	4	4	4	1
$\Sigma_{n,6}$	5	5	5	5	5	2

Table 4.1: n-horizon sensor schedules over set S, involving only one sensor.

The schedule integer values in Table 4.1 refer to the indices of the ordered set S.

Tables 4.2 and 4.3 present the outputs of numerical implementation of the algorithm where different combinations of sensors are used to construct sensor schedules.

S_o	α_i	N_o
$S_o[0] =$ $[3.0332, -5.1353, 2.6428, 0, 3]^T$	$\alpha_1 =$ $[1,2,3,0,0]^T$	3
$S_o[1] =$ $[5.1047, -22.2857, 0.8571, 8, 4]^T$	$\alpha_2 =$ $[0,0,4,8,0]^T$	1
$S_o[2] =$ $[0.6, 3, 0, 0, 0]^T$	$\alpha_3 =$ $[0,0,0,0,3]^T$	1

Table 4.2: Sensor vectors of the set S_o and their corresponding coordinate vector α_i.

	σ_0	σ_1	σ_2	σ_3	σ_4	$\text{rank}(\phi_n)$
$\Sigma_{n,1}$	0	0	0	1	2	5
$\Sigma_{n,2}$	1	2	0	0	0	5
$\Sigma_{n,3}$	0	1	0	0	2	5
$\Sigma_{n,4}$	2	1	0	0	0	5
$\Sigma_{n,5}$	0	0	1	1	2	5
$\Sigma_{n,6}$	0	1	0	1	2	5
$\Sigma_{n,7}$	0	2	0	1	1	5

$\Sigma_{n,8}$	1	2	1	0	0	5
$\Sigma_{n,9}$	1	2	0	1	2	4
$\Sigma_{n,10}$	2	0	2	1	2	3
$\Sigma_{n,11}$	2	0	1	1	2	4

Table 4.3: n-horizon sensor schedules over the set S_o and their corresponding observability matrix rank.

Note that the schedule integer values in Table 4.3 refer to the indices of the ordered set S_o.

Example 4.2. Consider the following sensor set S, where S does not cover the basis G.

$$S = \{S[0] = [-0.7953, -0.4210, 2, 0, 0]^T,$$
$$S[1] = [1.3333, -16, 0, 8, 0]^T,$$
$$S[2] = [1.1, -3, 0, 3, 0]^T,$$
$$S[3] = [1.1666, -14, 0, 7, 0]^T,$$
$$S[4] = [1.2, 6, 0, 0, 0]^T,$$
$$S[5] = [-1.1929, -0.6315, 3, 0, 0]^T\}.$$

Table 4.4 shows that no n-horizon observable sensor schedules can be constructed with only one sensor.

	σ_0	σ_1	σ_2	σ_3	σ_4	$\mathrm{rank}(\phi_n)$
$\Sigma_{n,1}$	0	0	0	0	0	1
$\Sigma_{n,2}$	1	1	1	1	1	1
$\Sigma_{n,3}$	2	2	2	2	2	2
$\Sigma_{n,4}$	3	3	3	3	3	1
$\Sigma_{n,5}$	4	4	4	4	4	1
$\Sigma_{n,6}$	5	5	5	5	5	1

Table 4.4: n-horizon sensor schedules over the set S involving only one sensor.

Note that the schedule integer values in Table 4.4 refer to the indices of the ordered set S.

Tables 4.5 and 4.6 present the outputs of numerical implementation of the algorithm where different combinations of sensors are used to construct sensor schedules.

S_o	α_i	N_o
$S_o[0] = [1.1, -3, 0, 3, 0]^T$	$\alpha_0 = [0,0,0,3,3]^T$	2
$S_o[1] = [-0.7953, -0.4210, 2, 0, 0]^T$	$\alpha_1 = [0,2,0,0,0]^T$	1
$S_o[2] = [1.3333, -16, 0, 8, 0]^T$	$\alpha_2 = [0,0,0,8,0]^T$	0
$S_o[3] = [1.1666, -14, 0, 7, 0]^T$	$\alpha_3 = [0,0,0,7,0]^T$	0
$S_o[4] = [1.2, 6, 0, 0, 0]^T$	$\alpha_4 = [0,0,0,0,6]^T$	0
$S_o[5] = [-1.1929, -0.6315, 3, 0, 0]^T$	$\alpha_5 = [0,3,0,0,0]^T$	0

Table 4.5: Sensor vectors of the set S_o and their corresponding coordinate vector α_i

	σ_0	σ_1	σ_2	σ_3	σ_4	$\mathrm{rank}(\phi_n)$
$\Sigma_{n,1}$	0	0	1	2	3	3
$\Sigma_{n,2}$	0	0	1	3	4	3
$\Sigma_{n,3}$	0	0	1	4	5	3
$\Sigma_{n,4}$	0	0	1	5	6	3
$\Sigma_{n,5}$	5	0	2	4	3	3
$\Sigma_{n,6}$	3	4	2	0	3	2
$\Sigma_{n,7}$	5	5	1	3	3	2
$\Sigma_{n,8}$	2	1	1	4	2	3
$\Sigma_{n,9}$	2	0	4	0	3	2

Table 4.6: n-horizon sensor schedules and their corresponding observability matrix rank.

Note that each integer value in Table 4.6 refers to an index of the ordered set S_o.

4.2. Discussion and verification

We can see that schedule $\Sigma_{n,1}$ of Table 4.3 supports the main result. Building a schedule using sensors vectors of S_o and following the proposed schedule structure of Theorem 3.1, a full rank observability matrix for schedule $\Sigma_{n,1}$ is obtained.

Looking at the rest of the schedules in Table 4.3, the first thing that we notice is that the proposed schedule structure of Theorem 3.1 is not the only schedule structure that can provide observability. This can be seen in schedules $\Sigma_{n,5}$, $\Sigma_{n,6}$, $\Sigma_{n,7}$ and $\Sigma_{n,8}$ in Table 4.3, which may suggest that there is a more general schedule structure that can determine observable schedules. The second noticeable behavior is the change of time steps. It seems that if a schedule is observable, changing the time steps of the sensors or in other words rotating the schedule elements does not disrupt the observability. This behavior can be seen in schedules $\Sigma_{n,1}$ to $\Sigma_{n,4}$ and $\Sigma_{n,5}$ to $\Sigma_{n,8}$ of Table 4.3.

Schedules $\Sigma_{n,9}$ to $\Sigma_{n,11}$ in Table 4.3 show that even if all sensors of S_o are used in the schedule, there is no guarantee that the schedule will provide observability unless the proposed schedule structure of Theorem 3.1 is used.

Concerning Example 4.2, Table 4.6 suggests that, regardless of what combination of sensors is used, there is no combination of sensors that can provide an observable n-horizon schedule. In this specific example, the condition considered in Theorem 3.1 turns out to be also necessary for the existence of a n-horizon sensor schedule. This remark raises a point that will be the subject of future investigation.

5. Conclusion and future developments

In this thesis, the concept of sensor scheduling was introduced. The observability of systems under scheduled measurements was investigated, resulting in the introduction of the concept of measurement time horizon and extension of the definition of observability to the case of linear discrete-time systems with time-variant measurements.

The conditions guaranteeing the existence of an observable schedule were investigated, and new definitions and results were introduced to provide a simplified proof of the theorem in [1] and a clearer characterization of the proposed theorem for the definition of an observable sensor schedule. In particular, an algorithm was devised capable of defining a set of sensors that constitute an observable sensor schedule of a specific structure.

The algorithm was numerically implemented. The numerical results suggest that a more general structure of an observable schedule could exist. This can be a subject of future development.

In this work, all conditions were investigated for sensors with $1 \times n$ measurement matrices. A future development of this work can also consist in including conditions supporting the existence of observable schedules for sensors with measurement matrices with a higher dimension.

Another topic for future work can be the application of a Kalman filtering approach with scheduled measurements. To do so, we need to develop algorithms that take into consideration both the observability and estimation error in the optimization problem. When developing these algorithms, the computational costs of these algorithms are an important factor to take into consideration. For example, the offline determination of the sensor set and identification of all observable sensor subsets can be considered as the first step for the development of this approach, then the choice between observable sensor schedules can be taken (possibly online) based on the criterion of minimizing the variance of the estimation error.

References

[1] Ilkturk, U. (2015). Observability methods in sensor scheduling. Arizona State University.

[2] Kalman, R. E. (1960, August). On the general theory of control systems. In Proceedings First International Conference on Automatic Control, Moscow, USSR (pp. 481-492).

[3] Kalman, R. E. (1963). Mathematical description of linear dynamical systems. Journal of the Society for Industrial and Applied Mathematics, Series A: Control, 1(2), 152-192.

[4] Vitus, M. P., Zhang, W., Abate, A., Hu, J., & Tomlin, C. J. (2012). On efficient sensor scheduling for linear dynamical systems. Automatica, 48(10), 2482-2493.

Acknowledgements

I want to express my deepest thanks to my advisers, professor Alessandro Colombo and Marcello Farina, who helped and guided me through every step of this thesis. Without their help and deep knowledge and understanding of the topic, this work would have not been possible.

I must also thank the polytechnic university of Milan for providing me with an environment to learn and grow.

Finally, I would like to thank my parents for all their love and support throughout this journey, without which none of this would have been possible. I am very grateful to all the friends and people always offering support and love.