

Kalman Filtering for General Discrete-time Linear Systems

Ramine Nikoukhah, Stephen L. Campbell, François Delebecque

► **To cite this version:**

Ramine Nikoukhah, Stephen L. Campbell, François Delebecque. Kalman Filtering for General Discrete-time Linear Systems. [Research Report] RR-3343, INRIA. 1998. <inria-00073346>

HAL Id: inria-00073346

<https://hal.inria.fr/inria-00073346>

Submitted on 24 May 2006

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Kalman filtering for general discrete-time linear systems

Ramine Nikoukhah , Stephen L. Campbell , François Delebecque

N° 3343

Janvier 1998

————— THÈME 4 —————

 ***Rapport
de recherche***

Kalman filtering for general discrete-time linear systems

Ramine Nikoukhah , Stephen L. Campbell* , François Delebecque

Thème 4 — Simulation et optimisation
de systèmes complexes
Projet META2

Rapport de recherche n° 3343 — Janvier 1998 — 24 pages

Abstract: Recursive state estimation problems for explicit and implicit time-invariant linear systems, both for systems with and without unknown inputs, can be formulated as a single problem usually referred to as descriptor Kalman filtering. Solutions to this problem have been proposed in the literature, however, these solutions either neglect possible contributions of future dynamics to the current estimate or make unnecessary assumptions on the structure of the system. In this paper we propose a solution to this problem which leads to a constructive method lifting these unnecessary assumptions. This method uses a generalization of the shuffle algorithm.

Key-words: Kalman filtering. Linear systems. Singular systems. Stochastic systems. Descriptor systems.

(Résumé : tsvp)

Research supported in part by the National Science Foundation under DMS-9423705, ECS-9500589, and INT-9605114.

* Dept. of Mathematics, North Carolina State University, Raleigh, NC 27695-8205. USA.

Unité de recherche INRIA Rocquencourt
Domaine de Voluceau, Rocquencourt, BP 105, 78153 LE CHESNAY Cedex (France)
Téléphone : (33) 01 39 63 55 11 – Télécopie : (33) 01 39 63 53 30

Filtrage de Kalman de systèmes linéaires à temps discret généraux

Résumé : Les problèmes d'estimation récursive de l'état de systèmes LTI implicites et explicites peuvent être formulés dans le cadre unique du filtrage de Kalman des systèmes descripteurs, aussi bien pour les systèmes à entrées connues que inconnues.

Différentes solutions à ce problème ont été proposées dans la littérature mais ces solutions ou bien ne prennent pas en compte les contributions des dynamiques futures pour l'estimation courante ou bien font des hypothèses restrictives inutiles sur la structure du système.

Dans ce papier, on propose une solution constructive qui élimine ces hypothèses restrictives. La méthode s'appuie sur une généralisation de l'algorithme de "shuffle".

Mots-clé : Filtrage de Kalman. Systèmes linéaires. Systèmes singuliers. Systèmes stochastiques. Systèmes descripteurs.

Contents

1	Introduction	3
2	Problem formulation	4
3	Preliminaries	6
3.1	Simplifying Assumptions	6
3.2	Well-posedness and consistency	7
4	Regular estimation problems	8
4.1	Regularity	8
4.2	Regularization	10
4.2.1	The stochastic shuffle algorithm	11
4.2.2	Regularization using stochastic shuffle algorithm	12
5	Kalman filter construction	14
6	Example	18
7	Conclusion	20
A	Scilab Functions	20
B	Scilab sample session	22

1 Introduction

There has been a considerable amount of research on the problem of Kalman filtering for linear time invariant systems. In the “standard” (explicit dynamics and no unknown input) case, the problem is well understood and numerically robust algorithms have been proposed. In the presence of descriptor (implicit) dynamics, however, the Kalman filtering problem becomes more complex and different problem formulations have been proposed in the literature.

One of the features of descriptor systems is that future dynamics can affect the present values of the state. But in most formulations of the Kalman filtering problem for descriptor systems, the estimate is constructed solely based on past dynamics and observations (see for example [1, 3, 6, 7]). This is done by making restrictive assumptions on the structure of the system such as Y -observability. The solution is then constructed by analogy with the non-descriptor case where future dynamics do not contribute to the estimate of the current state. The role of future dynamics is considered in [13] as an a-posteriori correction to the estimate based on past dynamics and observations. The drawback is that the method works only if the state is estimable based on past dynamics and observations alone, a condition which is not necessary. Another drawback of this method is that it uses polynomial matrix manipulations and is not numerically tractable.

The following example illustrates how future dynamics can affect the present estimate.

Example 1.1 *The following is a simple, well defined, stochastic descriptor system,*

$$\begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} x_1(k+1) \\ x_2(k+1) \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} x_1(k) \\ x_2(k) \end{pmatrix} + \begin{pmatrix} 0 \\ \alpha \end{pmatrix} w(k), \quad k \geq 0, \quad (1.1)$$

where w is a sequence of independent, zero-mean unit-variance Gaussian random variables. Consider the observations

$$y(k) = x_2(k) + r(k), \quad k \geq 0, \quad (1.2)$$

where r is a sequence of independent, zero-mean unit-variance Gaussian random variables, independent of w . A Kalman filter for this system corresponds to the recursive construction of

$$\begin{pmatrix} \hat{x}_1(i) \\ \hat{x}_2(i) \end{pmatrix} = \mathcal{E} \left\{ \begin{pmatrix} x_1(i) \\ x_2(i) \end{pmatrix} \middle| y(1), \dots, y(i) \right\} \quad (1.3)$$

where \mathcal{E} denotes the expected value.

Clearly in this case $x_1(i)$ is independent of observations $y(0)$ through $y(i)$. This means that conditioned on these observations, $x_1(i)$ remains zero-mean and with variance α^2 (note that $x_1(i) = -\alpha w(i+1)$). Thus

$$\hat{x}_1(i) = 0. \quad (1.4)$$

It is straightforward to show that

$$\hat{x}_2(i) = \frac{\alpha^2}{1 + \alpha^2} y(i), \quad (1.5)$$

and that the error covariance matrix

$$P_i = \mathcal{E} \left\{ \begin{pmatrix} x_1(i) - \hat{x}_1(i) \\ x_2(i) - \hat{x}_2(i) \end{pmatrix} \begin{pmatrix} x_1(i) - \hat{x}_1(i) \\ x_2(i) - \hat{x}_2(i) \end{pmatrix}^T \right\} = \begin{pmatrix} \alpha^2 & 0 \\ 0 & \frac{\alpha^2}{1+\alpha^2} \end{pmatrix}. \quad (1.6)$$

This is a well-posed estimation problem. However, it cannot be solved using the methods presented in any of the literature cited above since it does not satisfy the assumptions made in those papers. We shall come back to this problem later in Section 6.

The method presented in this paper is based on a recursive restructuring algorithm for transforming the estimation problem into one where future dynamics do not affect present state estimates (in which case we say that the problem is *regular*). We call this algorithm the *stochastic shuffle algorithm* because it can be considered as a generalization of the shuffle algorithm [10] to the stochastic case. It can also be considered as an extension of the structure algorithms of [14] and [9].

The Kalman filtering problem considered in this paper is formulated in Section 2. Section 3 contains some preliminary results. The stochastic shuffle algorithm is derived and used to show how a general estimation problem can be converted to a regular estimation problem in Section 4. The complete solution to the descriptor Kalman filtering problem is given in Section 5. Section 6 illustrates the method on a simple example.

2 Problem formulation

The descriptor Kalman filtering problem is usually formulated as follows:

$$Jx(k+1) = Ax(k) + u(k) + Bw(k) \quad (2.1)$$

$$y(k) = Cx(k) + Dr(k). \quad (2.2)$$

where w and r are independent sequences of independent zero-mean unit-covariance Gaussian vectors, y is the sequence of the measured output vectors, u is a known sequence of vectors, and $Vx(0)$, a projection of the initial condition, is assumed to be a Gaussian random vector with mean v_0 and covariance Q , independent of w and r . The objective is to construct, recursively, either the filtered estimate

$$\hat{x}(k) = \mathcal{E}\{x(k) \mid y(i), 0 \leq i \leq k\} \quad (2.3)$$

or the predicted estimate

$$\hat{x}(k) = \mathcal{E}\{x(k) \mid y(i), 0 \leq i \leq k-1\}. \quad (2.4)$$

It has been shown in [13] that this problem is equivalent to a maximum-likelihood (ML) estimation problem where the dynamics (2.1) and the a-priori information on $x(0)$ are reformulated as observations:

$$u(k) = Jx(k+1) - Ax(k) - Bw(k) \quad (2.5)$$

$$v_0 = Vx(0) + \tilde{v} \quad (2.6)$$

where \tilde{v} is Gaussian zero-mean with covariance Q and independent of w 's and r 's. x is now considered as a sequence of unknown parameters to be estimated using the ML approach.

The ML formulation also allows more generality. For example, the pencil $\{J, A\}$ can be non-regular or even non-square. Such cases arise, for example, when a system with unknown inputs is reformulated as a descriptor system by augmenting its state with its unknown inputs [5].

The general problem of Kalman filtering can now be formulated as follows:

$$G\nu(k) + L\eta(k) = E\xi(k+1) - F\xi(k) + H\omega(k), \quad k \geq 0, \quad (2.7)$$

$$\mu = K\xi(0) + M\zeta \quad (2.8)$$

where ω is a sequence of independent zero-mean unit-covariance Gaussian vectors, ζ is a zero-mean unit-covariance Gaussian vector independent of ω 's; η 's are known for all the time but only past values of ν are supposed to be known. The problem is to construct, by a recursive method, the ML estimate of ξ at every time k :

$$\hat{\xi}(i) = \mathcal{E}\{\xi(i) \mid \nu(k), 0 \leq k \leq i-1; \eta(k), 0 \leq k\}. \quad (2.9)$$

Specifically, $\hat{\xi}(i)$ is the ML estimate of $\xi(i)$ based on observations:

$$\mu = K\xi(0) + M\zeta \quad (2.10)$$

$$G\nu(k) + L\eta(k) = E\xi(k+1) - F\xi(k) + H\omega(k), \quad 0 \leq k \leq i-1, \quad (2.11)$$

$$L\eta(k) = -G\nu(k) + E\xi(k+1) - F\xi(k) + H\omega(k), \quad k \geq i. \quad (2.12)$$

Because of the Gaussian assumption, the ML estimation problem is equivalent to an LQ optimization problem. In particular, $\hat{\xi}(i)$ is the solution of the following optimization problem:

$$\mathcal{J} = \min_{\substack{\zeta, \xi, \omega, \eta \\ \nu(k), k \geq i}} \zeta^T \zeta + \sum_{j=0}^{\infty} \omega^T(j) \omega(j), \quad \text{subject to (2.10), (2.11) and (2.12)}. \quad (2.13)$$

A non-recursive solution to the ML problem would be to solve a new optimization problem at every k . This clearly is not an acceptable solution.

The solution of the general problem begins in Section 3. But first, we note that both the filtered and predicted estimation problems (2.1)-(2.2) can be reformulated as special cases of the ML problem (2.7)-(2.8). To see this in the predicted case, let (2.7) be reformulated as follows:

$$G = \begin{pmatrix} 0 \\ I \end{pmatrix}, \quad L = \begin{pmatrix} -I \\ 0 \end{pmatrix}, \quad E = \begin{pmatrix} -J \\ 0 \end{pmatrix}, \quad (2.14)$$

$$F = -\begin{pmatrix} A \\ C \end{pmatrix}, \quad H = \begin{pmatrix} B & 0 \\ 0 & D \end{pmatrix}, \quad (2.15)$$

and for $k \geq 0$, let

$$\nu(k) = y(k), \quad \eta(k) = u(k), \quad \xi(k) = x(k), \quad \omega(k) = \begin{pmatrix} w(k) \\ r(k) \end{pmatrix} \quad (2.16)$$

with (2.8) defined as

$$\mu = v_0, \quad K = V, \quad M = Q^{\frac{1}{2}}. \quad (2.17)$$

Here the square root of a matrix Q is any matrix M such that $MM^T = Q$.

In the filtered case, for $k \geq 0$, let (2.7) be reformulated as

$$G = \begin{pmatrix} 0 \\ I \end{pmatrix}, \quad L = \begin{pmatrix} -I \\ 0 \end{pmatrix}, \quad E = \begin{pmatrix} -J \\ C \end{pmatrix}, \quad (2.18)$$

$$F = -\begin{pmatrix} A \\ 0 \end{pmatrix}, \quad H = \begin{pmatrix} B & 0 \\ 0 & D \end{pmatrix}. \quad (2.19)$$

$$\nu(k) = y(k+1), \quad \eta(k) = u(k), \quad \xi(k) = x(k), \quad \omega(k) = \begin{pmatrix} w(k) \\ r(k+1) \end{pmatrix} \quad (2.20)$$

with (2.8) defined as

$$\mu = \begin{pmatrix} x_0 \\ y(0) \end{pmatrix}, \quad K = \begin{pmatrix} I \\ C \end{pmatrix}, \quad M = \begin{pmatrix} Q^{\frac{1}{2}} & 0 \\ 0 & D \end{pmatrix}. \quad (2.21)$$

If part, or all, of u is not known in the future, then it should be formulated as part of ν and not η . The distinction between ν and η has to do with their availability, or lack of it, in the future, and not their input-output nature. For example, if (2.1) was replaced with

$$Jx(k+1) = Ax(k) + S_1u_1(k) + S_2u_2(k) + Bw(k) \quad (2.22)$$

where the input u_1 is supposed known for all times, but only past values of u_2 are supposed known, then (2.7)-(2.8) should be constructed by setting (in the filtered case)

$$\nu(k) = \begin{pmatrix} u_2(k) \\ y(k+1) \end{pmatrix}, \quad \eta(k) = u_1(k). \quad (2.23)$$

3 Preliminaries

3.1 Simplifying Assumptions

Without any loss of generality we shall make the following assumptions on system (2.7)-(2.8):

- Assumption 1:** H has full column rank.
Assumption 2: $(E \ G \ H \ L)$ has full row rank.

Assumption 1 is not in any way restrictive. If H is not full column rank, we can do a QR decomposition:

$$H = \begin{pmatrix} H' & 0 \end{pmatrix} \begin{pmatrix} Q_1 \\ Q_2 \end{pmatrix} \quad (3.1)$$

where H' has full column rank and $\begin{pmatrix} Q_1 \\ Q_2 \end{pmatrix}$ is orthogonal. Then let

$$w'(k) = Q_1 w(k). \quad (3.2)$$

Clearly w' is a sequence of independent zero-mean unit-covariance Gaussian vectors because the rows of Q_1 are orthogonal to each other and have unit norm. Thus we can replace $Hw(k)$ by $H'w'(k)$.

As for Assumption 2, first note that without any loss of generality we can assume that $(E \ F \ G \ H \ L)$ has full row rank. In that case, if Assumption 2 is not satisfied, then a part of ξ is identically zero and can be discarded, using an extension of the projection algorithm [11], as follows:

- row-compress $(E \ G \ H \ L)$ by left-multiplication with an invertible matrix:

$$T = \begin{pmatrix} T_1 \\ T_2 \end{pmatrix} \quad (3.3)$$

so that

$$(E_1 \ G_1 \ H_1 \ L_1) = T_1(E \ G \ H \ L) \quad (3.4)$$

has full row rank and

$$T_2(E \ G \ H \ L) = 0. \quad (3.5)$$

- Let

$$\begin{pmatrix} F_1 \\ F_2 \end{pmatrix} = \begin{pmatrix} T_1 \\ T_2 \end{pmatrix} F \quad (3.6)$$

and do a QR decomposition of F_2 :

$$F_2 = \begin{pmatrix} F_{21} & 0 \end{pmatrix} \begin{pmatrix} S_1 \\ S_2 \end{pmatrix} \quad (3.7)$$

where F_{21} has full column rank and $\begin{pmatrix} S_1 \\ S_2 \end{pmatrix}$ is orthogonal. Let

$$\begin{cases} \xi_1(k) & = & S_1 \xi(k) \\ \xi_2(k) & = & S_2 \xi(k) \end{cases} . \quad (3.8)$$

- Noting that ξ_1 is identically zero since $F_{21}\xi_1 = 0$, (2.7)-(2.8) can be simplified to

$$G_1 \nu(k) + L_1 \eta(k) = E_1 \xi_2(k+1) - F_1 \xi_2(k) + H_1 \omega(k), \quad k \geq 0, \quad (3.9)$$

$$\mu = K_2 \xi_2(0) + \zeta \quad (3.10)$$

where

$$K_2 = K S_2^T. \quad (3.11)$$

- If $(E_1 \ G_1 \ H_1 \ L_1)$ has full row rank, we are done. Otherwise, we repeat this process.

This algorithm necessarily ends in a finite number of steps because at each step, the number of rows is reduced by at least one.

3.2 Well-posedness and consistency

The estimation problem (2.7)-(2.8) has a very general structure. In the absence of additional assumptions, this formulation may lead to contradictions. For example the system: $\eta(k) = \omega(k)$ does not make any sense because it would mean that the random sequence ω is equal for all times to the (a-priori) known sequence η . Such problems of course don't come up when system (2.7)-(2.8) is constructed from a reasonable estimation problem (as done earlier in this paper).

Definition 3.1 *The estimation problem (2.7)-(2.8) is called well-posed if for all ζ and all sequences η and ω , there exist a μ and sequences ν and ξ such that (2.7)-(2.8) is satisfied.*

Lemma 3.1 *The estimation problem (2.7)-(2.8) is well-posed if and only if $(zE - F \ G)$ has full generic row rank.*

Proof Since μ is unconstrained, (2.8) does not constrain ζ , η and ω and hence (2.8) does not affect whether the system is well-posed. Equation (2.7) can be rewritten using the forward shift operator z as

$$-H\omega + L\eta = (zE - F)\xi - G\nu. \quad (3.12)$$

All the constraints implied by (3.12) are obtained by premultiplication with a polynomial vector in z . If $(zE - F \ G)$ has full generic row rank, no constraint is implied on ω and η because this implies that there exists no vector $v(z) \neq 0$ such that

$$v(z)(zE - F \ G) = 0. \quad (3.13)$$

On the other hand, suppose there exists a $v(z) \neq 0$ satisfying (3.13) and that ω and η are completely arbitrary. Then from (3.12) we have

$$v(z)(-H\omega + L\eta) = 0 \quad (3.14)$$

which implies, since ω and η are completely arbitrary, that

$$v(z)(H \ L) = 0. \quad (3.15)$$

Let

$$v(z) = \sum_{i=0}^m v_m z^m \quad (3.16)$$

with $v_m \neq 0$. Then from (3.13) and (3.15) it follows that

$$v_m (E G H L) = 0 \quad (3.17)$$

which is a contradiction since $(E G H L)$ has full row rank. \square

Definition 3.2 Let (2.7)-(2.8) be well-posed. Then observations (μ, ν, η) are said to be consistent if there exist (ζ, ω, ξ) such that (2.7)-(2.8) is satisfied.

Clearly any well-posed system has at least one consistent set of observations.

4 Regular estimation problems

4.1 Regularity

Problems in which future dynamics do not contribute to current state estimates play an important role in our development. We call these problems regular.

Definition 4.1 The estimation problem which consists of constructing, for all $i \geq 0$, $\hat{\xi}(i)$ as the ML estimate of $\xi(i)$ based on (2.10) and (2.11) is called the auxiliary estimation problem associated with the estimation problem (2.7)-(2.8).

The Kalman filtering methods presented in the literature consider this auxiliary estimation problem, at least as a first step.

Definition 4.2 The estimation problem (2.7)-(2.8) is called regular if it is well-posed and, for all consistent (μ, η, ν) , its solution (the estimate $\hat{\xi}$, or the set of estimates if not unique) is equal to the solution of the associated auxiliary estimation problem.

Lemma 4.1 The estimation problem (2.7)-(2.8) is regular if $(E G)$ has full row rank.

Proof Suppose $(E G)$ has full row rank. We need to show that (2.12) does not contribute to the estimate of $\xi(i)$. Consider the optimization problem

$$\mathcal{J} = \min_{\phi} \min_{\substack{\zeta, \xi, \omega, \eta \\ \nu(k), k \geq i}} \zeta^T \zeta + \sum_{j=0}^{\infty} \omega^T(j) \omega(j), \quad (4.1)$$

subject to (2.10), (2.11), (2.12) and

$$\phi = \xi(i). \quad (4.2)$$

The optimization problem (4.1) has the same solution as the optimization problem (2.13) because the optimal ϕ in (4.1) is clearly equal to the optimal $\xi(i)$ in (2.13). The conditioning in ϕ however allows us to decouple the optimization problem as

$$\mathcal{J} = \min_{\phi} \mathcal{J}_p(\phi) + \mathcal{J}_f(\phi) \quad (4.3)$$

where

$$\mathcal{J}_p(\phi) = \min_{\substack{\zeta, \xi(l), l \leq i \\ \omega(k), \eta(k), \nu(k), k < i}} \zeta^T \zeta + \sum_{j=0}^{i-1} \omega^T(j) \omega(j), \quad \text{subject to (2.10), (2.11) and (4.2)} \quad (4.4)$$

and

$$\mathcal{J}_f(\phi) = \min_{\substack{\xi^{(k)}, \omega^{(k)}, \eta^{(k)}, \nu^{(k)} \\ k \geq i}} \sum_{j=i}^{\infty} \omega^T(j) \omega(j), \quad \text{subject to (2.12) and (4.2)}. \quad (4.5)$$

The optimization problem (4.4) corresponds to the auxiliary estimation problem. So to show regularity, it suffices to show that, regardless of the value of ϕ ,

$$\mathcal{J}_f(\phi) = 0. \quad (4.6)$$

This can be done by rewriting (4.5) as

$$\mathcal{J}_f(\phi) = \min_{\substack{\omega^{(k)}, \eta^{(k)}, \nu^{(k)}, k \geq i \\ \xi^{(l)}, l > i}} \sum_{j=i}^{\infty} \omega^T(j) \omega(j) \quad (4.7)$$

subject to

$$\mathcal{L} \begin{pmatrix} \eta^{(i)} \\ \eta^{(i+1)} \\ \eta^{(i+2)} \\ \vdots \end{pmatrix} = \begin{pmatrix} E & -G & & & \\ -F & 0 & E & -G & \\ & & -F & 0 & \ddots \\ & & & & \ddots & \ddots \end{pmatrix} \begin{pmatrix} \xi^{(i+1)} \\ \nu^{(i)} \\ \xi^{(i+2)} \\ \nu^{(i+1)} \\ \vdots \end{pmatrix} + \begin{pmatrix} -F \\ 0 \\ 0 \\ 0 \\ \vdots \end{pmatrix} \phi + \mathcal{M} \begin{pmatrix} \omega^{(i)} \\ \omega^{(i+1)} \\ \omega^{(i+2)} \\ \vdots \end{pmatrix} \quad (4.8)$$

where

$$\mathcal{L} = \text{diag}(L, L, L, \dots), \quad \mathcal{M} = \text{diag}(M, M, M, \dots). \quad (4.9)$$

Since $(E \ G)$ is full row rank, we have that the matrix

$$\begin{pmatrix} E & -G & & & \\ -F & 0 & E & -G & \\ & & -F & 0 & \ddots \\ & & & & \ddots & \ddots \end{pmatrix}$$

in (4.8) is onto as a linear transformation between vector spaces of sequences. Thus, for ω 's equal to zero and any ϕ , we can find

$$\begin{pmatrix} \xi^{(i+1)} \\ \nu^{(i)} \\ \xi^{(i+2)} \\ \nu^{(i+1)} \\ \vdots \end{pmatrix}$$

such that (4.8) is satisfied. This would clearly correspond to the optimal solution because it yields $\mathcal{J}_f(\phi) = 0$. \square

Note that a standard explicit Kalman filtering problem always leads to a regular estimation problem because in that case

$$E = \begin{pmatrix} I \\ 0 \end{pmatrix}, \quad G = \begin{pmatrix} 0 \\ I \end{pmatrix} \quad (4.10)$$

in the predicted formulation and

$$E = \begin{pmatrix} I \\ C \end{pmatrix}, \quad G = \begin{pmatrix} 0 \\ I \end{pmatrix} \quad (4.11)$$

in the filtered case. In both cases, $(E \ G)$ has full row rank. Thus there is no distinction between the estimation problem and the associated auxiliary problem in the standard explicit Kalman filtering problem.

4.2 Regularization

Definition 4.3 *Two well-posed estimation problems of type (2.7)-(2.8) are called equivalent if they have identical solution sets.*

We will show that if the estimation problem (2.7)-(2.8) is well-posed, then it has an equivalent regular estimation problem. We prove this using a constructive method based on an extension of the shuffle algorithm called the stochastic shuffle algorithm. Let us start with a simple example to illustrate the idea behind this method.

Example 4.1 *Consider the estimation problem*

$$\nu(k) = \xi(k+1) + \omega(k), \quad k \geq 0, \quad (4.12)$$

$$\eta(k) = \xi(k) + \omega(k), \quad k \geq 0, \quad (4.13)$$

$$\mu = \xi(0) + \zeta. \quad (4.14)$$

The corresponding E and G matrices are both equal to $(1 \ 0)^T$ and thus this problem is not regular. This can easily be seen by noting that, (4.13), for $k = i$, contributes to the estimation of $\xi(i)$ because η is known for all times. One way to solve this problem is to rewrite (4.13) as

$$\eta(k+1) = \xi(k+1) + \omega(k+1), \quad k \geq 0, \quad (4.15)$$

which of course is completely equivalent except for the very first equation which can be incorporated in the initial condition. The result is

$$\nu(k) = \xi(k+1) + \omega(k), \quad k \geq 0, \quad (4.16)$$

$$\eta(k+1) = \xi(k+1) + \omega(k+1), \quad k \geq 0, \quad (4.17)$$

$$\begin{pmatrix} \mu \\ \eta(0) \end{pmatrix} = \begin{pmatrix} 1 \\ 1 \end{pmatrix} \xi(0) + \begin{pmatrix} \zeta \\ \omega(0) \end{pmatrix}. \quad (4.18)$$

Now the $(E \ G)$ for (4.16)-(4.18) has full row rank but the problem is not in the standard form (2.7)-(2.8). Even if we let

$$\omega'(k) = \omega(k+1), \quad (4.19)$$

there remains the problem that

$$\begin{pmatrix} \omega(k) \\ \omega'(k) \end{pmatrix}$$

is not an independent sequence. So let us proceed in a slightly different manner. In particular, let us first rewrite (4.12)-(4.14) as

$$\nu(k) - \eta(k) = \xi(k+1) - \xi(k), \quad k \geq 0, \quad (4.20)$$

$$\eta(k) = \xi(k) + \omega(k), \quad k \geq 0, \quad (4.21)$$

$$\mu = \xi(0) + \zeta. \quad (4.22)$$

(4.20) is obtained simply by subtracting (4.13) from (4.12). Then we do the transformation (4.15). The result is now

$$\nu(k) - \eta(k) = \xi(k+1) - \xi(k), \quad k \geq 0, \quad (4.23)$$

$$\eta'(k) = \xi(k+1) + \omega'(k), \quad k \geq 0, \quad (4.24)$$

$$\begin{pmatrix} \mu \\ \eta(0) \end{pmatrix} = \begin{pmatrix} 1 \\ 1 \end{pmatrix} \xi(0) + \begin{pmatrix} \zeta \\ \omega(0) \end{pmatrix} \quad (4.25)$$

where ω' is defined in (4.19) and

$$\eta'(k) = \eta(k+1). \quad (4.26)$$

This problem is in the form (2.7)-(2.8), and is regular.

4.2.1 The stochastic shuffle algorithm

Consider the matrices E, F, G, H, L, K and M in (2.7)-(2.8).

0- Set $\kappa = 0$, and initialize the array of system matrices

$$\mathcal{S}_0 = E \mid G \mid F \mid H \mid L$$

and the array of initial condition matrices

$$\mathcal{T}_0 = K \mid M \mid I .$$

1- At the κ 'th iteration, consider the array of system matrices

$$\mathcal{S}_\kappa = \bar{E} \mid \bar{G} \mid \bar{F} \mid \bar{H} \mid \bar{L}_0 \mid \cdots \mid \bar{L}_\kappa$$

and the array of initial condition matrices

$$\mathcal{T}_\kappa = \bar{K} \mid \bar{M}_0 \quad \bar{N}_0 \mid \cdots \mid \bar{M}_\kappa \quad \bar{N}_\kappa .$$

If $(\bar{E} \quad \bar{G})$ has full row rank, stop. Otherwise, perform row operations to obtain

$$\mathcal{S}_\kappa = \begin{array}{c|c|c|c|c|c|c} E_1 & G_1 & F_1 & H_1 & L_{0,1} & \cdots & L_{\kappa,1} \\ 0 & 0 & F_2 & H_2 & L_{0,2} & \cdots & L_{\kappa,2} \end{array}$$

where $(E_1 \quad G_1)$ has full row rank and $H_1 H_2^T = 0$.

2- Do a “shuffle” as follows:

$$\mathcal{S}_{\kappa+1} = \begin{array}{c|c|c|c|c|c|c|c} E_1 & G_1 & F_1 & H_1 & L_{0,1} & L_{1,1} & \cdots & L_{\kappa,1} \\ -F_2 & 0 & 0 & H_2 & 0 & L_{0,2} & \cdots & L_{\kappa-1,2} \end{array} \mid \begin{array}{c} 0 \\ L_{\kappa,2} \end{array}$$

and update the array of initial condition matrices:

$$\mathcal{T}_{\kappa+1} = \begin{array}{c|c|c|c|c|c|c|c|c} \bar{K} & \bar{M}_0 & \bar{N}_0 & \bar{M}_1 & \bar{N}_1 & \cdots & \bar{M}_\kappa & \bar{N}_\kappa & 0 \\ -F_2 & 0 & 0 & 0 & L_{0,2} & \cdots & 0 & L_{\kappa-1,2} & H_2 \end{array} \mid \begin{array}{c} 0 \\ L_{\kappa,2} \end{array}$$

3- Set $\kappa = \kappa + 1$, and go back to step 1.

The key difference between the stochastic shuffle algorithm and the standard shuffle algorithm is in step 2, where in addition to row compressing $(E \quad G)$, we also require that

$$H_1 H_2^T = 0. \quad (4.27)$$

This, in particular, implies that $H_1 \omega(k)$ and $H_2 \omega(k)$ are stochastically independent.

Lemma 4.2 *It is always possible by row operations to row compress $(E \quad G)$ and simultaneously impose (4.27).*

Proof Let

$$U = \begin{pmatrix} U_1 \\ U_2 \end{pmatrix} \quad (4.28)$$

be any invertible matrix such that

$$U(E \quad G) = \begin{pmatrix} E_1 & G_1 \\ 0 & 0 \end{pmatrix} \quad (4.29)$$

where $(E_1 \quad G_1)$ has full row rank. Let T be any solution of

$$T(U_2 H H^T U_2^T) = -U_1 H H^T U_2^T. \quad (4.30)$$

This equation always has a solution because

$$\ker(U_2 H H^T U_2^T) = \ker(H^T U_2^T) \subset \ker(U_1 H H^T U_2^T). \quad (4.31)$$

Then

$$V = \begin{pmatrix} U_1 + T U_2 \\ U_2 \end{pmatrix} \quad (4.32)$$

is invertible,

$$V \begin{pmatrix} E & G \end{pmatrix} = \begin{pmatrix} E_1 & G_1 \\ 0 & 0 \end{pmatrix}, \quad (4.33)$$

and H_1 and H_2 defined as

$$\begin{pmatrix} H_1 \\ H_2 \end{pmatrix} = V H \quad (4.34)$$

satisfy (4.27) because

$$H_1 H_2^T = (U_1 + T U_2) H H^T U_2^T = 0. \quad (4.35)$$

Thus the row operations corresponding to left multiplication with V perform the desired operations. \square

Lemma 4.2 shows that all the steps of the stochastic shuffle algorithm can be implemented. The following result shows that the algorithm ends in a finite number of steps.

Lemma 4.3 *If $(zE - F \ G)$ has full generic row rank, the stochastic shuffle algorithm ends in less than n iterations where n denotes the size of ξ .*

Proof After each iteration, the system is premultiplied by an invertible matrix, which performs the row operations, and the matrix

$$\begin{pmatrix} I & 0 \\ 0 & zI \end{pmatrix}$$

where z denotes the forward shift operator. The identity matrices are possibly each of different size at each iteration. Thus after m iterations, for some invertible matrices V_1 through V_m , we have

$$(z\bar{E} - \bar{F} \ \bar{G}) = \begin{pmatrix} I & 0 \\ 0 & zI \end{pmatrix} V_m \begin{pmatrix} I & 0 \\ 0 & zI \end{pmatrix} \cdots V_1 (zE - F \ G) \quad (4.36)$$

Since $(zE - F \ G)$ has full row rank (the well-posedness assumption), we can find a constant matrix W such that $(zE - F \ G)W$ is invertible for all except possibly a finite number of z values. By postmultiplying with W and taking the determinant of both sides of (4.36), we get on the left hand side a polynomial the degree of which is bounded above by the rank \bar{E} . On the right hand side is a polynomial whose degree is at least $m + \text{rank}(E)$. Thus $m \leq n - \text{rank}(E) \leq n$. \square

4.2.2 Regularization using stochastic shuffle algorithm

We can now use the results of the stochastic shuffle algorithm applied to the well-posed estimation problem (2.7)-(2.8),

$$\mathcal{S}_\kappa = \{ \bar{E} \ \bar{G} \ \bar{F} \ \bar{H} \ \bar{L}_0 \ \cdots \ \bar{L}_\kappa \} \quad (4.37)$$

$$\mathcal{T}_\kappa = \{ \bar{K} \ \bar{M}_0 \ \bar{N}_0 \ \cdots \ \bar{M}_\kappa \ \bar{N}_\kappa \} \quad (4.38)$$

to construct an equivalent regular estimation problem. That is, to *regularize* it.

Theorem 4.1 Let (4.37), (4.38) be the matrices at the κ -th step of the stochastic shuffle algorithm. Consider the estimation problem

$$\bar{G}\nu(k) + (\bar{L}_0 \cdots \bar{L}_\kappa) \begin{pmatrix} \eta(k) \\ \vdots \\ \eta(k + \kappa) \end{pmatrix} = \bar{E}\xi(k + 1) - \bar{F}\xi(k) + \bar{H}\omega'(k), \quad k \geq 0, \quad (4.39)$$

$$(\bar{N}_0 \bar{N}_1 \cdots \bar{N}_\kappa) \begin{pmatrix} \mu \\ \eta(0) \\ \vdots \\ \eta(\kappa - 1) \end{pmatrix} = \bar{K}\xi(0) + (\bar{M}_0 \cdots \bar{M}_\kappa) \zeta' \quad (4.40)$$

where ω' is a sequence of independent zero-mean unit-covariance Gaussian vectors and ζ' is a zero-mean unit-covariance Gaussian vector independent of ω' . Then the estimation problems (2.7)-(2.8) and (4.39)-(4.40) are equivalent.

Proof After the first step of the stochastic shuffle algorithm, (2.7) becomes

$$\begin{pmatrix} G_1 \\ 0 \end{pmatrix} \nu(k) + \begin{pmatrix} L_1 \\ L_2 \end{pmatrix} \eta(k) = \begin{pmatrix} E_1 \\ 0 \end{pmatrix} \xi(k + 1) - \begin{pmatrix} F_1 \\ F_2 \end{pmatrix} \xi(k) + \begin{pmatrix} H_1 \\ H_2 \end{pmatrix} \omega(k), \quad k \geq 0. \quad (4.41)$$

Clearly this system is equivalent to (2.7) because we simply have done left multiplication by an invertible matrix. By changing k with $k + 1$ in the subsystem appearing at the bottom of (4.41), we get

$$\begin{pmatrix} G_1 \\ 0 \end{pmatrix} \nu(k) + \begin{pmatrix} L_1\eta(k) \\ L_2\eta(k + 1) \end{pmatrix} = \begin{pmatrix} E_1 \\ -F_2 \end{pmatrix} \xi(k + 1) - \begin{pmatrix} F_1 \\ 0 \end{pmatrix} \xi(k) + \begin{pmatrix} H_1\omega(k) \\ H_2\omega(k + 1) \end{pmatrix}, \quad k \geq 0. \quad (4.42)$$

Note that (4.42) has the same equations as (4.41) except that it is missing

$$L_2\eta(0) = -F_2\xi(0) + H_2\omega(0). \quad (4.43)$$

But this equation can be included in the initial condition as follows:

$$\begin{pmatrix} \mu \\ L_2\eta(0) \end{pmatrix} = \begin{pmatrix} K \\ -F_2 \end{pmatrix} \xi(0) + \begin{pmatrix} M & 0 \\ 0 & H_2\omega(0) \end{pmatrix} \begin{pmatrix} \zeta \\ \omega(0) \end{pmatrix}. \quad (4.44)$$

Thus the system is still equivalent to the original system but it is no longer in the original format because $\omega(k)$ and $\omega(k + 1)$ both appear in the equation. Let ω' be a sequence given by

$$\omega'(k) = \begin{pmatrix} H_1 \\ H_2 \end{pmatrix}^+ \begin{pmatrix} H_1\omega(k) \\ H_2\omega(k + 1) \end{pmatrix} \quad (4.45)$$

where $(\cdot)^+$ denotes the left-inverse which exists in this case thanks to Assumption 1. Then

$$\begin{pmatrix} H_1 \\ H_2 \end{pmatrix} \omega'(k) = \begin{pmatrix} H_1\omega(k) \\ H_2\omega(k + 1) \end{pmatrix} \quad (4.46)$$

because, thanks to (4.27),

$$\text{Im} \begin{pmatrix} H_1 \\ H_2 \end{pmatrix} = \text{Im} \begin{pmatrix} H_1 & 0 \\ 0 & H_2 \end{pmatrix}. \quad (4.47)$$

In order to complete the proof of Theorem 4.1, we need the following lemma.

Lemma 4.4 ω' is a sequence of independent zero-mean unit-covariance Gaussian vectors.

Proof of Lemma 4.4 From (4.45), we have that $\omega'(k)$'s are zero-mean and Gaussian. Since ω is an independent sequence, it is straightforward to show that

$$\forall k, j \text{ such that } |k - j| \geq 2, \quad \mathcal{E} \{ \omega'(k) \omega'(j)^T \} = 0. \quad (4.48)$$

From (4.46), we get

$$\begin{pmatrix} H_1 \\ H_2 \end{pmatrix} \mathcal{E} \{ \omega'(k) \omega'(k)^T \} \begin{pmatrix} H_1 \\ H_2 \end{pmatrix}^T = \begin{pmatrix} H_1 H_1^T & 0 \\ 0 & H_2 H_2^T \end{pmatrix}. \quad (4.49)$$

Thanks to (4.27),

$$\mathcal{E} \{ \omega'(k) \omega'(k)^T \} = I \quad (4.50)$$

satisfies (4.49), and it is the unique solution thanks to Assumption 1. Moreover, from (4.46) and (4.27), we get

$$\begin{pmatrix} H_1 \\ H_2 \end{pmatrix} \mathcal{E} \{ \omega'(k) \omega'(k+1)^T \} \begin{pmatrix} H_1 \\ H_2 \end{pmatrix}^T = \begin{pmatrix} H_1 & 0 \\ 0 & H_2 \end{pmatrix} \begin{pmatrix} 0 & 0 \\ I & 0 \end{pmatrix} \begin{pmatrix} H_1^T & 0 \\ 0 & H_2^T \end{pmatrix} = 0. \quad (4.51)$$

Thus the $\omega'(k)$'s are zero-mean, unit-covariance Gaussian and independent. \square

Returning to the proof of Theorem 4.1 and using (4.46), the system can be expressed as

$$\begin{pmatrix} G_1 \\ 0 \end{pmatrix} \nu(k) + \begin{pmatrix} L_1 & 0 \\ 0 & L_2 \end{pmatrix} \begin{pmatrix} \eta(k) \\ \eta(k+1) \end{pmatrix} = \begin{pmatrix} E_1 \\ -F_2 \end{pmatrix} \xi(k+1) - \begin{pmatrix} F_1 \\ 0 \end{pmatrix} \xi(k) + \begin{pmatrix} H_1 \\ H_2 \end{pmatrix} \omega'(k), \quad k \geq 0. \quad (4.52)$$

$$\begin{pmatrix} I & 0 \\ 0 & L_2 \end{pmatrix} \begin{pmatrix} \mu \\ \eta(0) \end{pmatrix} = \begin{pmatrix} K \\ -F_2 \end{pmatrix} \xi(0) + \begin{pmatrix} M & 0 \\ 0 & H_2 \end{pmatrix} \begin{pmatrix} \zeta \\ \omega(0) \end{pmatrix}. \quad (4.53)$$

which corresponds exactly to step 2 of the algorithm. Thus after one iteration of the stochastic shuffle algorithm, we end up with an equivalent estimation problem in the same format. The proof then follows by induction. \square

5 Kalman filter construction

As we have shown in the previous section, any well-posed system can be regularized, so, without any loss of generality we can assume that the estimation problem (2.7)-(2.8) is regular.

Definition 5.1 *If the estimation problem (2.7)-(2.8) has a unique solution, we say that ξ is causally estimable.*

Lemma 5.1 *Let the estimation problem (2.7)-(2.8) be regular. Then ξ is causally estimable if and only if E and K both have full column rank.*

Proof Thanks to the regularity assumption, the solution of the estimation problem (2.7)-(2.8) is given by the ML estimate of $\xi(i)$ based on (2.10) and (2.11) which can be rewritten as

$$\begin{pmatrix} \mu \\ G\nu(0) + L\eta(0) \\ \vdots \\ G\nu(i-1) + L\eta(i-1) \end{pmatrix} = \begin{pmatrix} K & & & \\ -F & E & & \\ & \ddots & \ddots & \\ & & -F & E \end{pmatrix} \begin{pmatrix} \xi(0) \\ \xi(1) \\ \vdots \\ \xi(i) \end{pmatrix} + \begin{pmatrix} M\zeta \\ H\omega(0) \\ \vdots \\ H\omega(i-1) \end{pmatrix}. \quad (5.1)$$

Clearly if E and K both have full column rank, so does

$$\begin{pmatrix} K & & & \\ -F & E & & \\ & \ddots & \ddots & \\ & & -F & E \end{pmatrix}$$

and thus $\xi(i)$ is estimable.

On the other hand if K does not have full column rank and $\hat{\xi}(0)$ is an ML estimate of $\xi(0)$, $\hat{\xi}(0) + \gamma$ is also an ML estimate of $\xi(0)$ where $\gamma \in \ker K$. If E does not have full column rank and $\hat{\xi}(i)$ is an ML estimate of $\xi(i)$, for any $i \geq 1$, $\hat{\xi}(i) + \gamma$ is also an ML estimate of $\xi(i)$ where $\gamma \in \ker E$. In either case, ξ is not causally estimable. \square

Theorem 5.1 *Let the estimation problem (2.7)-(2.8) be regular and let ξ be causally estimable. Then*

$$\hat{\xi}(k+1) = \begin{pmatrix} 0 & I \end{pmatrix} \begin{pmatrix} FP_kF^T + HH^T & E \\ E^T & 0 \end{pmatrix}^\dagger \begin{pmatrix} F\hat{\xi}(k) + G\nu(k) + L\eta(k) \\ 0 \end{pmatrix}, \quad k \geq 0. \quad (5.2)$$

$$\hat{\xi}(0) = \begin{pmatrix} 0 & I \end{pmatrix} \begin{pmatrix} MM^T & K \\ K^T & 0 \end{pmatrix}^\dagger \begin{pmatrix} \mu \\ 0 \end{pmatrix} \quad (5.3)$$

where P_k denotes the error covariance matrix:

$$P_k = \mathcal{E} \left\{ (\xi(k) - \hat{\xi}(k))(\xi(k) - \hat{\xi}(k))^T \right\} \quad (5.4)$$

and is given iteratively by

$$P_{k+1} = - \begin{pmatrix} 0 & I \end{pmatrix} \begin{pmatrix} FP_kF^T + HH^T & E \\ E^T & 0 \end{pmatrix}^\dagger \begin{pmatrix} 0 \\ I \end{pmatrix}, \quad k \geq 0. \quad (5.5)$$

$$P_0 = - \begin{pmatrix} 0 & I \end{pmatrix} \begin{pmatrix} MM^T & K \\ K^T & 0 \end{pmatrix}^\dagger \begin{pmatrix} 0 \\ I \end{pmatrix}. \quad (5.6)$$

Here the pseudo-inverse Z^\dagger of a symmetric matrix Z is any symmetric matrix satisfying

$$ZZ^\dagger Z = Z \quad (5.7)$$

(This is a special kind of what is referred to in [4] as a (1)-inverse).

Note that thanks to the regularity assumption, the estimation problem (2.7)-(2.8) is equivalent to its associated auxiliary problem. This latter problem is considered in [13]. Theorem 5.1 is a straightforward extension of Theorem 3.1 of [13] and the proof is very similar.

Theorem 5.2 *Suppose*

$$\text{Left-ker}((sE - tF \ H)) \text{ is independent of } s \text{ and } t, \forall (s, t) \neq (0, 0) \text{ such that } |s| \geq |t|, \quad (5.8)$$

$$sE - tF \text{ has full column rank } \forall (s, t) \neq (0, 0) \text{ such that } |s| \geq |t|. \quad (5.9)$$

Then P_k in (5.5) converges exponentially to the unique positive semi-definite solution P of the algebraic Riccati equation

$$P = - \begin{pmatrix} 0 & I \end{pmatrix} \begin{pmatrix} FPF^T + HH^T & E \\ E^T & 0 \end{pmatrix}^\dagger \begin{pmatrix} 0 \\ I \end{pmatrix}. \quad (5.10)$$

Moreover, the resulting filter

$$\hat{\xi}(k+1) = \begin{pmatrix} 0 & I \end{pmatrix} \begin{pmatrix} FPF^T + HH^T & E \\ E^T & 0 \end{pmatrix}^\dagger \begin{pmatrix} F\hat{\xi}(k) + G\nu(k) + L\eta(k) \\ 0 \end{pmatrix} \quad (5.11)$$

is stable.

This theorem is proved in [13] (Theorem 4.3) under the more stringent condition

$$(sE - tF \ H) \text{ has full row rank } \forall (s, t) \neq (0, 0) \text{ such that } |s| \geq |t|, \quad (5.12)$$

instead of (5.8).

Proof Note that if (5.8) holds, then the constant left kernel has to be the left kernel of $(E \ F \ H)$. Thus if $(E \ F \ H)$ has full row rank, we would be back under assumption (5.12). Suppose then that $(E \ F \ H)$ is not full row rank. Let

$$T = \begin{pmatrix} T_1 \\ T_2 \end{pmatrix} \quad (5.13)$$

be an orthogonal matrix that row compresses $(E \ F \ H)$. That is,

$$(E_1 \ F_1 \ H_1) = T_1 (E \ F \ H) \quad (5.14)$$

has full row rank and

$$T_2 (E \ F \ H) = 0. \quad (5.15)$$

We now need the following lemma.

Lemma 5.2 *The matrices E_1 , F_1 and H_1 from (5.14) satisfy*

$$(sE_1 - tF_1 \ H_1) \text{ has full row rank } \forall (s, t) \neq (0, 0) \text{ such that } |s| \geq |t|, \quad (5.16)$$

$$sE_1 - tF_1 \text{ has full column rank } \forall (s, t) \neq (0, 0) \text{ such that } |s| \geq |t|. \quad (5.17)$$

Proof of Lemma 5.2 Note that

$$T(sE - tF) = \begin{pmatrix} sE_1 - tF_1 \\ 0 \end{pmatrix} \quad (5.18)$$

so clearly (5.17) follows (5.9).

Suppose $(sE_1 - tF_1 \ H_1)$ does not have full row rank $\forall (s, t) \neq (0, 0)$, $|s| \geq |t|$. In that case, there exists $(s_0, t_0) \neq (0, 0)$, $|s_0| \geq |t_0|$, and $p_1 \neq 0$ such that

$$p_1 (s_0 E_1 - t_0 F_1 \ H_1) = 0. \quad (5.19)$$

Using the fact that

$$T_1 T_1^T = I \quad (5.20)$$

we get

$$p_1 T_1 T_1^T (s_0 E_1 - t_0 F_1 \ H_1) = p (s_0 E - t_0 F \ H) = 0 \quad (5.21)$$

where

$$p = p_1 T_1 \neq 0 \quad (5.22)$$

since T_1 has full row rank. But then thanks to Assumption (5.8), (5.21) implies that

$$p (sE - tF \ H) = 0, \quad \forall (s, t). \quad (5.23)$$

Thus

$$p_1 (sE_1 - tF_1 \ H_1) = 0, \quad \forall (s, t) \quad (5.24)$$

which in turn implies that

$$p_1 (E_1 \ F_1 \ H_1) = 0. \quad (5.25)$$

But this is a contradiction because $(E_1 \ F_1 \ H_1)$ is full row rank. \square

Returning to the proof of Theorem 5.2, if

$$\begin{pmatrix} X & Y \\ Y^T & Z \end{pmatrix} = \begin{pmatrix} F P F^T + H H^T & E \\ E^T & 0 \end{pmatrix}^\dagger, \quad (5.26)$$

then

$$\begin{pmatrix} F_1 P F_1^T + H_1 H_1^T & E_1 \\ E_1^T & 0 \end{pmatrix}^\dagger = \begin{pmatrix} T_1 X T_1^T & T_1 Y \\ Y^T T_1^T & Z \end{pmatrix}. \quad (5.27)$$

To see this, simply note that

$$\begin{pmatrix} F P F^T + H H^T & E \\ E^T & 0 \end{pmatrix} \begin{pmatrix} X & Y \\ Y^T & Z \end{pmatrix} \begin{pmatrix} F P F^T + H H^T & E \\ E^T & 0 \end{pmatrix} = \begin{pmatrix} F P F^T + H H^T & E \\ E^T & 0 \end{pmatrix} \quad (5.28)$$

which, after pre and post-multiplication by

$$\begin{pmatrix} T_1 & 0 \\ 0 & I \end{pmatrix} \quad \text{and} \quad \begin{pmatrix} T_1^T & 0 \\ 0 & I \end{pmatrix}$$

and, using (5.14) and

$$E = T_1^T E_1, \quad F = T_1^T F_1, \quad H = T_1^T H_1, \quad (5.29)$$

implies that

$$\begin{pmatrix} F_1 P F_1^T + H_1 H_1^T & E_1 \\ E_1^T & 0 \end{pmatrix} \begin{pmatrix} T_1 X T_1^T & T_1 Y \\ Y^T T_1^T & Z \end{pmatrix} \begin{pmatrix} F_1 P F_1^T + H_1 H_1^T & E_1 \\ E_1^T & 0 \end{pmatrix} = \begin{pmatrix} F_1 P F_1^T + H_1 H_1^T & E_1 \\ E_1^T & 0 \end{pmatrix}. \quad (5.30)$$

Thus

$$\begin{aligned} P_{k+1} &= - \begin{pmatrix} 0 & I \end{pmatrix} \begin{pmatrix} F P_k F^T + H H^T & E \\ E^T & 0 \end{pmatrix}^\dagger \begin{pmatrix} 0 \\ I \end{pmatrix} = Z \\ &= - \begin{pmatrix} 0 & I \end{pmatrix} \begin{pmatrix} F_1 P_k F_1^T + H_1 H_1^T & E_1 \\ E_1^T & 0 \end{pmatrix}^{-1} \begin{pmatrix} 0 \\ I \end{pmatrix}. \end{aligned} \quad (5.31)$$

The inverse exists in (5.31) because E_1 has full column rank, thanks to (5.16), and the fact that

$$(F_1 P_k F_1^T + H_1 H_1^T \quad E_1) \quad \text{has full row rank.} \quad (5.32)$$

Condition (5.32) holds because $(E_1 \ H_1)$ has full row rank, thanks to (5.16), and P_k is positive semi-definite.

Thanks to Lemma 5.2, Theorem 4.3 of [13] applies to (5.31) and P_k converges exponentially to the unique positive semi-definite solution of

$$P = - \begin{pmatrix} 0 & I \end{pmatrix} \begin{pmatrix} F_1 P F_1^T + H_1 H_1^T & E_1 \\ E_1^T & 0 \end{pmatrix}^{-1} \begin{pmatrix} 0 \\ I \end{pmatrix}. \quad (5.33)$$

But we have shown that

$$\begin{pmatrix} 0 & I \end{pmatrix} \begin{pmatrix} F P F^T + H H^T & E \\ E^T & 0 \end{pmatrix}^\dagger \begin{pmatrix} 0 \\ I \end{pmatrix} = \begin{pmatrix} 0 & I \end{pmatrix} \begin{pmatrix} F_1 P F_1^T + H_1 H_1^T & E_1 \\ E_1^T & 0 \end{pmatrix}^{-1} \begin{pmatrix} 0 \\ I \end{pmatrix}. \quad (5.34)$$

Thus P_k converges exponentially to the unique positive semi-definite solution P of (5.10).

To show stability, we need to show that the eigenvalues of $Y^T F$ are inside the unit circle. From Theorem 4.3 of [13], we know that the eigenvalues of $Y^T T_1^T F_1$ are inside the unit circle. But

$$Y^T F = Y^T T_1^T F_1. \quad (5.35)$$

This completes the proof. \square

The next example shows that condition (5.8) is weaker than condition (5.12).

Example 5.1 *The estimation problem*

$$\eta(k) = \xi(k+1) + \omega(k) \quad (5.36)$$

$$\nu(k) - \eta(k) = 0 \quad (5.37)$$

is well-posed, regular and ξ is causally estimable. The associated Kalman filter is clearly

$$\hat{\xi}(k+1) = \nu(k), \quad \text{or equivalently } \eta(k), \quad (5.38)$$

with associated error covariance matrix

$$P_k = P = 1. \quad (5.39)$$

In this case,

$$E = \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \quad F = \begin{pmatrix} 0 \\ 0 \end{pmatrix}, \quad H = \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \quad (5.40)$$

so that (5.8) is satisfied but not (5.12). Thus Theorem 5.2 does apply in this case.

The following result follows from (5.34), Lemma 5.2, and Theorem 5.1 of [13].

Theorem 5.3 *The matrix P in (5.10) can be constructed as follows:*

$$P = X(E_1^T Y)^{-1} \quad (5.41)$$

where the columns of $\begin{pmatrix} X \\ Y \end{pmatrix}$ form a basis for the stable eigenmodes of the regular pencil

$$\left\{ \begin{pmatrix} E_1 & -H_1 H_1^T \\ 0 & F_1^T \end{pmatrix}, \begin{pmatrix} F_1 & 0 \\ 0 & E_1^T \end{pmatrix} \right\}$$

and where E_1 , F_1 and H_1 are defined in (5.14).

Note that under the more restrictive assumption (5.12), we can let

$$E_1 = E, \quad F_1 = F, \quad H_1 = H. \quad (5.42)$$

6 Example

We return to Example 1.1 and add a known input u to make the problem more interesting.

$$\begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} x_1(k+1) \\ x_2(k+1) \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} x_1(k) \\ x_2(k) \end{pmatrix} + \begin{pmatrix} 0 \\ 1 \end{pmatrix} u(k) + \begin{pmatrix} 0 \\ \alpha \end{pmatrix} w(k), \quad (6.1)$$

$$y(k) = x_2(k) + r(k). \quad (6.2)$$

This problem can be formulated as in (2.7)-(2.8) by letting $\nu(k) = y(k+1)$, $\eta(k) = u(k)$, $\omega(k) = \begin{pmatrix} w(k) \\ r(k+1) \end{pmatrix}$, so that

$$G = \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}, \quad L = \begin{pmatrix} 0 \\ -1 \\ 0 \end{pmatrix}, \quad H = \begin{pmatrix} 0 & 0 \\ \alpha & 0 \\ 0 & 1 \end{pmatrix},$$

$$E = \begin{pmatrix} 0 & -1 \\ 0 & 0 \\ 0 & 1 \end{pmatrix}, \quad F = \begin{pmatrix} -1 & 0 \\ 0 & -1 \\ 0 & 0 \end{pmatrix}$$

with the initial condition $\mu = y(0)$, $K = (0 \ 1)$ and $M = 1$.

	E	G	F	H	L
$\kappa = 1$	$\begin{matrix} 0 & -1 \\ 0 & 0 \\ 0 & 1 \end{matrix}$	$\begin{matrix} 0 \\ 0 \\ 1 \end{matrix}$	$\begin{matrix} -1 & 0 \\ 0 & -1 \\ 0 & 0 \end{matrix}$	$\begin{matrix} 0 & 0 \\ \alpha & 0 \\ 0 & 1 \end{matrix}$	$\begin{matrix} 0 \\ -1 \\ 0 \end{matrix}$
Compress: Exchange rows 2 and 3	$\begin{matrix} 0 & -1 \\ 0 & 1 \\ 0 & 0 \end{matrix}$	$\begin{matrix} 0 \\ 1 \\ 0 \end{matrix}$	$\begin{matrix} -1 & 0 \\ 0 & 0 \\ 0 & -1 \end{matrix}$	$\begin{matrix} 0 & 0 \\ 0 & 1 \\ \alpha & 0 \end{matrix}$	$\begin{matrix} 0 \\ 0 \\ -1 \end{matrix}$
Shuffle $\kappa = 2$	$\begin{matrix} 0 & -1 \\ 0 & 1 \\ 0 & 1 \end{matrix}$	$\begin{matrix} 0 \\ 1 \\ 0 \end{matrix}$	$\begin{matrix} -1 & 0 \\ 0 & 0 \\ 0 & 0 \end{matrix}$	$\begin{matrix} 0 & 0 \\ 0 & 1 \\ \alpha & 0 \end{matrix}$	$\begin{matrix} 0 & 0 \\ 0 & 0 \\ 0 & -1 \end{matrix}$
Compress: Add row 1 to row 3	$\begin{matrix} 0 & -1 \\ 0 & 1 \\ 0 & 0 \end{matrix}$	$\begin{matrix} 0 \\ 1 \\ 0 \end{matrix}$	$\begin{matrix} -1 & 0 \\ 0 & 0 \\ -1 & 0 \end{matrix}$	$\begin{matrix} 0 & 0 \\ 0 & 1 \\ \alpha & 0 \end{matrix}$	$\begin{matrix} 0 & 0 \\ 0 & 0 \\ 0 & -1 \end{matrix}$
Shuffle $\kappa = 3$	$\begin{matrix} 0 & -1 \\ 0 & 1 \\ 1 & 0 \end{matrix}$	$\begin{matrix} 0 \\ 1 \\ 0 \end{matrix}$	$\begin{matrix} -1 & 0 \\ 0 & 0 \\ 0 & 0 \end{matrix}$	$\begin{matrix} 0 & 0 \\ 0 & 1 \\ \alpha & 0 \end{matrix}$	$\begin{matrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & -1 \end{matrix}$

Table 1: The evolution of \mathcal{S}_k

κ	K	M_0	N_0	M_1	N_1	M_2	N_2
1	$\begin{matrix} 0 & 1 \end{matrix}$	1	1				
2	$\begin{matrix} 0 & 1 \\ 0 & 1 \end{matrix}$	$\begin{matrix} 1 \\ 0 \end{matrix}$	$\begin{matrix} 1 \\ 0 \end{matrix}$	$\begin{matrix} 0 & 0 \\ \alpha & 0 \end{matrix}$	$\begin{matrix} 0 \\ -1 \end{matrix}$		
3	$\begin{matrix} 0 & 1 \\ 0 & 1 \\ 1 & 0 \end{matrix}$	$\begin{matrix} 1 \\ 0 \\ 0 \end{matrix}$	$\begin{matrix} 1 \\ 0 \\ 0 \end{matrix}$	$\begin{matrix} 0 & 0 \\ \alpha & 0 \\ 0 & 0 \end{matrix}$	$\begin{matrix} 0 \\ -1 \\ 0 \end{matrix}$	$\begin{matrix} 0 & 0 \\ 0 & 0 \\ \alpha & 0 \end{matrix}$	$\begin{matrix} 0 \\ 0 \\ -1 \end{matrix}$

Table 2: The evolution of \mathcal{T}_k

This system satisfies the assumptions but it is clearly not regular because $(E \ G)$ does not have full row rank. The steps of the stochastic shuffle algorithm are given in Tables 1 and 2.

Thus after regularization (and removing useless zero columns from the M matrix), we get

$$\begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix} y(k+1) + \begin{pmatrix} 0 \\ 0 \\ -1 \end{pmatrix} u(k+2) = \begin{pmatrix} 0 & -1 \\ 0 & 1 \\ 1 & 0 \end{pmatrix} \begin{pmatrix} x_1(k+1) \\ x_2(k+1) \end{pmatrix} - \begin{pmatrix} 0 & 1 \\ 0 & 0 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} x_1(k) \\ x_2(k) \end{pmatrix} + \begin{pmatrix} 0 & 0 \\ 0 & 1 \\ \alpha & 0 \end{pmatrix} \omega'(k), \quad k \geq 0, \quad (6.3)$$

$$\begin{pmatrix} y(0) \\ -u(1) \\ -u(2) \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ 0 & 1 \\ 1 & 0 \end{pmatrix} \begin{pmatrix} x_1(0) \\ x_2(0) \end{pmatrix} + \begin{pmatrix} 1 & 0 & 0 \\ 0 & \alpha & 0 \\ 0 & 0 & \alpha \end{pmatrix} \zeta'. \quad (6.4)$$

Using the results of Section 5, we can construct the corresponding Kalman filter:

$$\hat{x}_1(k+1) = -u(k+2) \quad (6.5)$$

$$\hat{x}_2(k+1) = \frac{1}{1+\alpha^2} \hat{x}_1(k) + \frac{\alpha^2}{1+\alpha^2} y(k+1) \quad (6.6)$$

with initial condition

$$\hat{x}_1(0) = -u(1) \quad (6.7)$$

$$\hat{x}_2(0) = -\frac{1}{1+\alpha^2} u(0) + \frac{\alpha^2}{1+\alpha^2} y(0) \quad (6.8)$$

and error covariance matrices

$$P_k = P = \begin{pmatrix} \alpha^2 & 0 \\ 0 & \frac{\alpha^2}{1+\alpha^2} \end{pmatrix}. \quad (6.9)$$

In the case $u = 0$, we are back to Example 1.1 and our result is consistent with the result we obtained in Section 1.

7 Conclusion

We have introduced in this paper a new structuring algorithm for stochastic descriptor systems. This algorithm (called the stochastic shuffle algorithm) converts the system, without changing some important statistical properties, to a system which has an important causality property. The stochastic shuffle algorithm is used to give a complete solution to the descriptor Kalman filtering problem.

The stochastic shuffle algorithm should have applications to other problems in the stochastic theory of descriptor systems, in particular the stochastic realization problem. This problem is currently under study.

Scilab implementation of the Stochastic Shuffle Algorithm

A Scilab Functions

Scilab function `ssalg` implements the stochastic shuffle algorithm. `ssalg` is a recursive function, i.e., it calls itself inside the function until $[E \ G]$ becomes full row rank.

If $[zE - F \ G]$ does not have full generic row rank (problem not well-posed) an error message is issued.

```
function [S,T]=ssalg(S,T)
//Stochastic shuffle Algorithm
//S,T:Scilab lists
[E,G,F,H]=S(1:4);
listL=S;for i=1:4;listL(i)=null();end;
K=T(1);listMN=T;listMN(1)=null();
kappa=length(listL)-1;
```

```

[nrE,ncE]=size(E);
if kappa>ncE then error('Problem not well-posed!');end
[X,H1,H2]=rcomp([E,G],H); //compression phase
if H2==[] then return;end
E=X*E;G=X*G;F=X*F;
for t=0:kappa
    listL(t+1)=X*listL(t+1);
end
k=size(H1,'r');
E1=E(1:k,:);E2=zeros(nrE-k,ncE);
G1=G(1:k,:);G2=zeros(nrE-k,size(G,'c'));
F1=F(1:k,:);F2=F(k+1:nrE,:);
E=[E1;-F2];G=[G1;G2];F=[F1;0*F2];H=[H1;H2];
//Start of shuffle phase
S=list(E,G,F,H);
S($+1)=[listL(1)(1:k,:);
        zeros(size(E,'r')-k,size(listL(1),'c'))];
for t=1:kappa
    S($+1)=[listL(t+1)(1:k,:);
            listL(t)(k+1:size(E,'r'),:)];
end
S($+1)=[zeros(k,size(listL(1),'c'));
        listL(kappa+1)(k+1:size(E,'r'),:)];
listup=listMN;
listup($+1)=zeros(size(listMN(1),'r'),size(H,'c'));
listup($+1)=zeros(listMN(2));
listdown=list();
for l=[1:2:length(listup)-3,2]
    listdown(l)=zeros(size(F2,'r'),size(listup(l),'c'));
end
for kk=1:kappa+1
    listdown(2*kk+2)=S(kk+5)(k+1:size(E,'r'),:);
end
listdown($-1)=H2;
T=list([K;-F2]);
for l=1:length(listup)
    T($+1)=[listup(l);listdown(l)];
end
//recursive call to ssalg
[S,T]=ssalg(S,T)

function [X,H1,H2]=rcomp(A,H)
//X rowcompresses A and [H1;H2]=X*H is such that H1*H2'=0;
if size(A,'r')~=size(H,'r') then
    error('A and H must have same number of rows!');
end
[X,k]=rowcomp(A); //k=rank(A)
U1=X(1:k,:);U2=X(k+1:$,:);
HH=H*H';T=-U1*HH*U2'*pinv(U2*HH*U2');
X=[U1+T*U2;U2];
XH=X*H;H1=XH(1:k,:);H2=XH(k+1:$,:);

```


B Scilab sample session

The following script illustrates the use of the `ssalg` function. It corresponds to the example treated in the paper. Note the result is the same as the result given in the paper modulo left multiplication by invertible matrices.

```

E=[0,-1;
  0,0;
  0,1];

F=[-1,0;
  0,-1;
  0,0];

G=[0;0;1];

L=[0;-1;0];

H=[0,0;
  alpha,0;
  0,1];

K=[0,1];M=1;

S=list(E,G,F,H,L);

T=list(K,M,1);

[S,T]=ssalg(S,T)

T =

      T(1)
!  0.          1. !
!  0.          1. !
!  0.7071068   0. !

      T(2)
!  1. !
!  0. !
!  0. !

      T(3)
!  1. !
!  0. !
!  0. !

      T(4)
!  0.  0. !
!  1.  0. !
!  0.  0. !

```

T(5)

```
!  0. !
! - 1. !
!  0. !
```

T(6)

```
!  0.      0. !
!  0.      0. !

!  0.7071068  0. !
```

T(7)

```
!  0.      !
!  0.      !
! - 0.7071068 !
```

S =

S(1)

```
!  0.      1.7071068 !
!  0.      0.2928932 !
!  0.7071068  0.      !
```

S(2)

```
!  0.7071068 !
! - 0.7071068 !
!  0.      !
```

S(3)

```
!  1.  0. !
!  1.  0. !
!  0.  0. !
```

S(4)

```
!  0.      0.7071068 !
!  0.      - 0.7071068 !
!  0.7071068  0.      !
```

S(5)

```
!  0. !
!  0. !
!  0. !
```

S(6)

```
!  0. !
!  0. !
!  0. !
```

S(7)

! 0. !
 ! 0. !
 ! - 0.7071068 !

References

- [1] Bernhard, P. and X. M. Wang (1987). Filtrage des systèmes implicites linéaires discrètes, *C. R. Acad. Sc. Paris*, t. **304**, Série I, n° **12**, 351-355.
- [2] Brennan, K. E. , S. L. Campbell and L. R. Petzold (1996). *The Numerical Solution of Initial Value Problems in Differential-Algebraic Equations*. SIAM.
- [3] Dai, L. (1989). Filtering and LQG problems for discrete-time stochastic singular systems. *IEEE Trans. on Automat. Control*, **AC-34**, 1105-1108.
- [4] Campbell, S. L. and C.D. Meyer, Jr. (1991). *Generalized Inverses of Linear Transformations*. Dover, New York.
- [5] Darouach, M., M. Zasadzinski, A. Basson Onana and S. Nowakowski (1995). Kalman filtering with unknown inputs via optimal state estimation of singular systems. *Int. J. Systems Sci.*, **Vol. 26**, **No. 10**, 2015-2028.
- [6] Darouach, M., M. Zasadzinski and D. Mehdi (1993). State estimation of stochastic singular linear systems. *Int. J. Systems Sci.*, **Vol. 2**, **No. 2**, 345-354.
- [7] Keller, J. Y., S. Nowakowski and M. Darouach (1992). State estimation and failure detection in singular systems. *Control-Theory and Advanced Technology*, **Vol. 8**, **No. 4**, 755-762.
- [8] Levy, B. C., A. Benveniste and R. Nikoukhah (1996), High-level primitives for recursive maximum likelihood estimation, *IEEE Trans. on Automat. Control*, **AC-41**, 1125-1145.
- [9] Lewis, F. L. (1990). Geometric design techniques for observers in singular systems. *Automatica*, **26**, 411-415.
- [10] Luenberger, D. G. (1978). Time-invariant descriptor systems. *Automatica*, **Vol. 14**, 473-480.
- [11] Nikoukhah, R., S. L. Campbell, F. Delebecque, On observer design for general linear time-invariant systems, *Automatica*, to appear.
- [12] Nikoukhah, R., D. Taylor, R., A. S. Willsky and B. C. Levy (1995). Graph Structure and Recursive Estimation of Noisy Linear Relations. *J. of Math. Systems, Estimation, and Control*, **5(4)**, 1-37.
- [13] Nikoukhah, R., A. S. Willsky and B. C. Levy (1992). Kalman filtering and Riccati equations for descriptor systems. *IEEE Trans. Automat. Control*, **AC-37**, 1325-1342.
- [14] Silverman, L. M. (1969). Inversion of multivariable linear systems. *IEEE Trans. Automat. Control*, **AC-14**, 270-276.



Unité de recherche INRIA Lorraine, Technopôle de Nancy-Brabois, Campus scientifique,
615 rue du Jardin Botanique, BP 101, 54600 VILLERS LÈS NANCY
Unité de recherche INRIA Rennes, Irista, Campus universitaire de Beaulieu, 35042 RENNES Cedex
Unité de recherche INRIA Rhône-Alpes, 655, avenue de l'Europe, 38330 MONTBONNOT ST MARTIN
Unité de recherche INRIA Rocquencourt, Domaine de Voluceau, Rocquencourt, BP 105, 78153 LE CHESNAY Cedex
Unité de recherche INRIA Sophia-Antipolis, 2004 route des Lucioles, BP 93, 06902 SOPHIA-ANTIPOLIS Cedex

Éditeur
INRIA, Domaine de Voluceau, Rocquencourt, BP 105, 78153 LE CHESNAY Cedex (France)
<http://www.inria.fr>
ISSN 0249-6399