ESAIM: COCV 17 (2011) 406–409
DOI: 10.1051/cocv/2011001

ESAIM: Control, Optimisation and Calculus of Variations
www.esaim-cocv.org

ERRATUM OF ARTICLE "REDUCED-ORDER UNSCENTED KALMAN FILTERING WITH APPLICATION TO PARAMETER IDENTIFICATION IN LARGE-DIMENSIONAL SYSTEMS"

Philippe Moireau[1] and Dominique Chapelle[1]

Received November 16, 2010.
Published online January 19, 2011.

ESAIM: COCV (2010) DOI: 10.1051/cocv/2010006

Some errors were introduced in (3.8) when summarizing the derivations of Section 3.2 (specifically in the second line of (3.8b) and third line of (3.8c)), hence we here rewrite the whole corrected summary for completeness.

Algorithm summary for the simplex case. Given adequate sampling rules, precompute the corresponding $[I^*]$,

- **Sampling**:

$$\begin{cases} C_n = \sqrt{U_n^{-1}} \\ \hat{X}_n^{(i)+} = \hat{X}_n^+ + L_n C_n I^{(i)}, \quad 1 \le i \le p+1. \end{cases} \tag{3.8a}$$

- **Prediction**:

$$\begin{cases} \hat{X}_{n+1}^- = E_\alpha(A(\hat{X}_n^{*+})) \\ \hat{X}_{n+1}^{(i)-} = \left| \begin{array}{ll} \hat{X}_{n+1}^- + [A(\hat{X}_n^{*+})] D_\alpha [V^*]^T ([V^*] D_\alpha [V^*]^T)^{-1/2} I^{(i)} & \text{with resampling} \\ \text{or} & \\ A(\hat{X}_n^{(i)+}) & \text{without resampling} \end{array} \right. \\ L_{n+1} = [X_{n+1}^{*-}] D_\alpha [V^*]^T \in \mathcal{M}_{d,p} \\ P_{n+1}^- = L_{n+1} (P_\alpha^V)^{-1} L_{n+1}^T. \end{cases} \tag{3.8b}$$

- **Correction**:

$$\begin{cases} Z_{n+1}^{(i)} = H(\hat{X}_{n+1}^{(i)-}), \\ \{HL\}_{n+1} = [Z_{n+1}^*] D_\alpha [V^*]^T \\ U_{n+1} = P_\alpha^V + \{HL\}_{n+1}^T W_{n+1}^{-1} \{HL\}_{n+1} \in \mathcal{M}_p \\ \hat{X}_{n+1}^+ = \hat{X}_{n+1}^- + L_{n+1} U_{n+1}^{-1} \{HL\}_{n+1}^T W_{n+1}^{-1} (Z_{n+1} - E_\alpha(Z_{n+1}^*)) \\ P_{n+1}^+ = L_{n+1} U_{n+1}^{-1} L_{n+1}^T. \end{cases} \tag{3.8c}$$

[1] INRIA, B.P. 105, 78153 Le Chesnay Cedex, France. philippe.moireau@inria.fr; dominique.chapelle@inria.fr

Next, although the other algorithm summary (3.14) was algebraically correct, it implicitly assumed the matrix $D_{\mathfrak{m}}$ to be invertible, which does not always hold. Hence, we rewrite the whole algorithm without using this assumption.

Algorithm summary for the general case. Given adequate sampling rules, precompute the corresponding $[V^*]$, $P_\alpha^{\scriptscriptstyle V} = [V^*]D_\alpha[V^*]^T$, $[I^*] = ([V^*]D_\alpha[V^*]^T)^{-\frac{1}{2}}[V^*]$, and $D_{\scriptscriptstyle V} = D_\alpha[V^*]^T(P_\alpha^{\scriptscriptstyle V})^{-1}[V^*]D_\alpha$.

- **Sampling**:

$$\begin{cases} C_n = \sqrt{U_n^{-1}} \\ \hat{X}_n^{(i)+} = \hat{X}_n^+ + L_n C_n I^{(i)}, \quad 1 \le i \le r. \end{cases} \tag{3.14a}$$

- **Prediction**:

$$\begin{cases} \hat{X}_{n+1}^- = E_\alpha(A(\hat{X}_n^{*+})) \\ \hat{X}_{n+1}^{(i)-} = \hat{X}_{n+1}^- + [A(\hat{X}_n^{*+}) - \hat{X}_{n+1}^-]D_\alpha^{\frac{1}{2}}\Upsilon_p I^{(i)}, \quad \text{resampling with SVD} \\ L_{n+1} = [X_{n+1}^{*-}]D_\alpha[V^*]^T \in \mathcal{M}_{d,p} \\ P_{n+1}^- = L_{n+1}(P_\alpha^{\scriptscriptstyle V})^{-1}L_{n+1}^T. \end{cases} \tag{3.14b}$$

- **Correction**:

$$\begin{cases} [\tilde{Z}] = [H(\hat{X}_{n+1}^*) - E_\alpha(H(\hat{X}_{n+1}^*))] \\ D_{\mathfrak{m}} = [\tilde{Z}]^T W_{n+1}^{-1}[\tilde{Z}] \in \mathcal{M}_r \\ U_{n+1} = P_\alpha^{\scriptscriptstyle V} + [V^*]D_\alpha\big(\mathbb{1} + D_{\mathfrak{m}}(D_\alpha - D_V)\big)^{-1}D_{\mathfrak{m}}D_\alpha[V^*]^T \in \mathcal{M}_p \\ \{HL\}_{n+1} = [\tilde{Z}](\mathbb{1} + D_\alpha D_{\mathfrak{m}})^{-1}\Big(\mathbb{1} + D_{\scriptscriptstyle V}\big(\mathbb{1} + D_{\mathfrak{m}}(D_\alpha - D_V)\big)^{-1}D_{\mathfrak{m}}\Big)D_\alpha[V^*]^T \\ \hat{X}_{n+1}^+ = \hat{X}_{n+1}^- + L_{n+1}U_{n+1}^{-1}\{HL\}_{n+1}^T W_{n+1}^{-1}(Z_{n+1} - E_\alpha(Z_{n+1}^*)) \\ P_{n+1}^+ = L_{n+1}U_{n+1}^{-1}L_{n+1}^T. \end{cases} \tag{3.14c}$$

We provide the proof for the correction step (3.14c) which contains the alternative equations valid without any assumption on $D_{\mathfrak{m}}$. First, we can write the filter in the form

$$\hat{K}_{n+1} = P_\alpha^{\tilde{x}\tilde{z}}(P_\alpha^{\tilde{z}})^{-1},$$

and we will compute this operator using the matrix inversion lemma to obtain a tractable algorithm. To this end we introduce the following compact notation

$$[\tilde{X}] = [\hat{X}_{n+1}^* - \hat{X}_{n+1}^-], \quad [\tilde{Z}] = [Z_{n+1}^* - E_\alpha(Z_{n+1}^*)],$$

and we then have

$$\begin{aligned} \hat{K}_{n+1} &= [\tilde{X}]D_\alpha[\tilde{Z}]^T(W_{n+1} + [\tilde{Z}]D_\alpha[\tilde{Z}]^T)^{-1} \\ &= [\tilde{X}]D_\alpha[\tilde{Z}]^T\Big(W_{n+1}^{-1} - W_{n+1}^{-1}[\tilde{Z}](D_\alpha^{-1} + [\tilde{Z}]^T W_{n+1}^{-1}[\tilde{Z}])^{-1}[\tilde{Z}]^T W_{n+1}^{-1}\Big) \\ &= [\tilde{X}]D_\alpha\Big(\mathbb{1}_r - [\tilde{Z}]^T W_{n+1}^{-1}[\tilde{Z}](D_\alpha^{-1} + [\tilde{Z}]^T W_{n+1}^{-1}[\tilde{Z}])^{-1}\Big)[\tilde{Z}]^T W_{n+1}^{-1}. \end{aligned}$$

Let us now set

$$D_{\mathfrak{m}} = [\tilde{Z}]^T W_{n+1}^{-1}[\tilde{Z}] \in \mathcal{M}_r,$$

which – unlike for $P_\alpha^{\tilde{z}}$ – can be computed in practice, since its dimension is equal to the number of sigma-points.

We thus have

$$\hat{K}_{n+1} = [\tilde{X}]D_\alpha(\mathbb{1} - D_{\mathfrak{m}}(D_\alpha^{-1} + D_{\mathfrak{m}})^{-1})[\tilde{Z}]^T W_{n+1}^{-1},$$
$$= [\tilde{X}]D_\alpha(\mathbb{1} + D_{\mathfrak{m}} D_\alpha)^{-1}[\tilde{Z}]^T W_{n+1}^{-1}, \quad (1)$$

where we have used the matrix inversion lemma in the second line. Note that the invertibility of a matrix $\mathbb{1} + AB$ with both A and B symmetric positive matrices is a standard property (*e.g.* one-to-one can be proven by decomposing $\mathbb{R}^r$ into the direct sum of Ker$A$ and ImA). Then, by the same argument as in Proposition 3.1, the filter can also be written in the form

$$\hat{K}_{n+1} = L_{n+1}(P_\alpha^{V})^{-1}[V^*]D_\alpha(\mathbb{1} + D_{\mathfrak{m}} D_\alpha)^{-1}[\tilde{Z}]^T W_{n+1}^{-1}, \quad (2)$$

with

$$L_{n+1} = [\hat{X}_{n+1}^{*-}]D_\alpha[V^*]^T.$$

Note that the term $[\tilde{Z}]^T$ in (1) cannot be treated in the same manner since the sigma-points propagated by the observation operator do not satisfy the original constraints. In addition to the gain, we also need to compute the *a posteriori* covariance matrix in order to resample at the next step. We have

$$P_{n+1}^{+} = P_{n+1}^{-} - P_\alpha^{\tilde{x}\tilde{z}}(P_\alpha^{\tilde{z}})^{-1}(P_\alpha^{\tilde{x}\tilde{z}})^T \quad (3)$$
$$= P_{n+1}^{-} - [\tilde{X}]D_\alpha\big(\mathbb{1} - D_{\mathfrak{m}}(D_\alpha^{-1} + D_{\mathfrak{m}})^{-1}\big)D_{\mathfrak{m}} D_\alpha[\tilde{X}]^T. \quad (4)$$

We now use the matrix inversion lemma as in (1) to simplify

$$\begin{aligned} P_{n+1}^{+} &= P_{n+1}^{-} - [\tilde{X}]D_\alpha(\mathbb{1} + D_{\mathfrak{m}} D_\alpha)^{-1}D_{\mathfrak{m}} D_\alpha[\tilde{X}]^T \\ &= P_{n+1}^{-} - L_{n+1}(P_\alpha^{V})^{-1}[V^*]D_\alpha(\mathbb{1} + D_{\mathfrak{m}} D_\alpha)^{-1}D_{\mathfrak{m}} D_\alpha[V^*]^T(P_\alpha^{V})^{-1}L_{n+1}^T \\ &= L_{n+1}\Big((P_\alpha^{V})^{-1} - (P_\alpha^{V})^{-1}[V^*]D_\alpha(\mathbb{1} + D_{\mathfrak{m}} D_\alpha)^{-1}D_{\mathfrak{m}} D_\alpha[V^*]^T(P_\alpha^{V})^{-1}\Big)L_{n+1}^T. \end{aligned}$$

The advantage of this last form is that we can again write

$$P_{n+1}^{+} = L_{n+1}U_{n+1}^{-1}L_{n+1}^T,$$

with

$$U_{n+1}^{-1} = (P_\alpha^{V})^{-1} - (P_\alpha^{V})^{-1}[V^*]D_\alpha(\mathbb{1} + D_{\mathfrak{m}} D_\alpha)^{-1}D_{\mathfrak{m}} D_\alpha[V^*]^T(P_\alpha^{V})^{-1}.$$

Hence, defining $D_V \in \mathcal{M}_r$ as

$$D_V = D_\alpha[V^*]^T(P_\alpha^{V})^{-1}[V^*]D_\alpha,$$

we can simplify – with another application of the matrix inversion lemma

$$U_{n+1} = P_\alpha^{V} + [V^*]D_\alpha\big(\mathbb{1} + D_{\mathfrak{m}}(D_\alpha - D_V)\big)^{-1}D_{\mathfrak{m}} D_\alpha[V^*]^T.$$

This identity of course requires that $\big(\mathbb{1} + D_{\mathfrak{m}}(D_\alpha - D_V)\big)$ be invertible, which can be established by proving that $D_\alpha - D_V$ is a symmetric positive matrix. We have by definition, indeed,

$$[V^*]D_V[V^*]^T = P_\alpha^{V}(P_\alpha^{V})^{-1}P_\alpha^{V} = [V^*]D_\alpha[V^*]^T,$$

hence, for any vector $R \in \mathbb{R}^r$ in the range of the rows of $[V^*]$

$$R^T(D_\alpha - D_V)R = 0.$$

If we now consider $S \in \mathbb{R}^r$ D_α-orthogonal to this range, namely, satisfying

$$[V^*]D_\alpha S = 0,$$

we have

$$S^T D_V S = S^T D_\alpha [V^*]^T (P_\alpha^V)^{-1} [V^*] D_\alpha S = 0,$$

hence,

$$S^T(D_\alpha - D_V)S = S^T D_\alpha S \geq 0,$$

which shows that $D_\alpha - D_V$ is positive as claimed.

It is now obvious that by defining

$$\{HL\}_{n+1} = [\tilde{Z}](\mathbb{1} + D_\alpha D_\mathfrak{m})^{-1}\Big(\mathbb{1} + D_V\big(\mathbb{1} + D_\mathfrak{m}(D_\alpha - D_V)\big)^{-1} D_\mathfrak{m}\Big) D_\alpha [V^*]^T,$$

we can rewrite the filter in the following form

$$\hat{X}_{n+1}^{+} = \hat{X}_{n+1}^{-} + L_{n+1} U_{n+1}^{-1} \{HL\}_{n+1}^T W_{n+1}^{-1} (Z_{n+1} - E_\alpha(Z_{n+1}^*)).$$

Finally, we point out that this algorithm is implemented in the Verdandi[2] opensource data assimilation library.

[2]http://verdandi.gforge.inria.fr/