AN ADAPTIVE FILTER USING A MOVING SENSOR FOR A CLASS OF DISTRIBUTED PARAMETER SYSTEMS

PIETRO MURACA* AND PAOLO PUGLIESE†

Abstract. An adaptive filter for a class of distributed parameter systems is presented, based on the use of a moving sensor, in order to obtain better reconstruction of the state and to reduce the number of convergence failures. The results, based up to now on numerical experiments, show good properties of convergence of the filter and a good reconstruction of the state.

Key words. Distributed parameter systems, adaptive filtering, on-line estimation.

1. Introduction. A relevant problem, when dealing with Distributed Parameter Systems (DPS), is the estimation of the state of the system when the exact values of some parameters are not known. The estimation process is based on output measurements taken by sensors located along the spatial domain of the system.

When the positions of the sensors can be chosen, naturally arises the problem of optimise their locations, according to some optimality criterion; this problem has been widely investigated in the past years. Most authors look for the optimal location of a set of fixed sensors; more recent approaches ([1], [2]) suggest the use of moving sensors, especially in those cases in which the system is time-varying.

In our work we propose to use a finite-dimensional Extended Kalman Filter (EKF) as an adaptive filter to estimate a finite-dimensional approximation of the state and the parameter of the DPS.

Such an approach is especially suited in case of on-line estimation, but it suffers from some problems, such as the divergence of the filter and the biasedness of the estimate [3]. In order to improve the performances of the filter, we use a moving sensor, whose position is chosen at any time so as to maximise the influence of the current observation on the variance of the extended state. This criterion has a plain stochastic interpretation in the case in which the parameters are known exactly, and only the state has to be estimated [4]. In our case the meaning of this approach is not obvious, but the algorithm which results shows good properties of convergence.

In all cases we have tested, the numerical experiences show that the proposed filter reconstructs the state of the system much better than an EKF with fixed sensor, for all the tested fixed location of the sensor; moreover it converges in all cases, whereas the EKF using a fixed sensor very often fails to converge.

2. Problem formulation. The class of systems we consider is described by the parabolic diffusion equation

$$\frac{\partial u(x,t)}{\partial t} = \frac{\partial}{\partial x}\left(a(x)\frac{\partial u(x,t)}{\partial x}\right) + f(x)w(t),$$

$x \in \Omega = [0;1], t \in [0;T_f]$, initial condition $u(x,0) = u_0(x), u_0 \in H_0^1(\Omega)$, boundary conditions $u(0,t) = u(1,t) = 0, t \in [0;T_f]$; $a(x) \in C(\Omega), f(x) \in L^2(\Omega), w(t) \in L^2(0;T_f)$. In order to ensure the stability of the DPS, the parameter $a(x)$ is assumed positive

* Dip. di Elettronica, Informatica e Sistemistica, Università della Calabria, 87036 Rende (Cs), Italy. {pietro@deis20.deis.unical.it}.

† Istituto per la Sistemistica e l' Informatica, CNR, c/o Dip. di Elettronica, Informatica e Sistemistica, Università della Calabria, 87036 Rende (Cs), Italy. {pugliese@deis20.deis.unical.it}.

for all x. In our approach the system is firstly reduced to a Lumped Parameter System by a Galerkin-type approximation procedure using the Finite Elements method [5] with piecewise linear basis function $\phi_i(\cdot)$; then we take as approximate solution $u_n(x,t) = \sum_1^n \phi_i(x) u_i(t)$; from the Galerkin method, functions $u_i(t)$ are solutions of

$$\text{(1)} \qquad \begin{cases} E\dot{\mathbf{u}}(t) &= A(\boldsymbol{\alpha})\mathbf{u}(t) + b\, w(t) \\ E\mathbf{u}(0) &= \mathbf{u}_0 \end{cases}$$

where $\mathbf{u}(t) = [u_1(t) \ldots u_n(t)]'$. Matrix E is nonsingular and depends only on the basis functions $\phi_i(\cdot)$; vector b depends both on $\phi_i(\cdot)$ and on the input distribution function $f(\cdot)$; $\mathbf{u}_0$ depends both on $\phi_i(\cdot)$ and on the initial condition $u_0(x)$; $A(\boldsymbol{\alpha})$ is symmetric, tridiagonal, negative definite if $a(x) > 0$; its nonzero entries are given by

$$A_{ij} = \begin{cases} -(\alpha_{i+1} + \alpha_{i+2}) & \text{if } j = i \\ \alpha_{i+1} & \text{if } j = i+1 \end{cases} \qquad \alpha_i = \frac{1}{h^2} \int_{x_{i-1}}^{x_i} a(x)\, dx$$

$i = 1, \ldots, n+1$; h is the length of each subinterval $[x_{i-1}; x_i]$, $i = 0, \ldots, n$, in which Ω is partitioned. Matrix A depends explicitly on the parameter vector $\boldsymbol{\alpha} \in \mathbb{R}^{n+1}$, which retains the information about the parameter function $a(x)$.

Assume that noisy measurements are taken, at discrete time instants t_k, by a sensor which may be located in a set of admissible observation points $\aleph$; we take $\aleph$ as the set of the internal nodes of the partition on Ω. Hence the observations are described by $y(t_k) = u(x^m(t_k), t_k) + v(t_k)$, $x^m(t_k)$ being the abscissa of the observation point at time t_k, and $v(\cdot)$ a Gaussian, discrete-time, zero-mean white noise with variance σ_V^2. By defining $C(k) = [C_1(k) \ldots C_n(k)]$, $C_i(k) = 1$ if $x^m(t_k)$ is the *i-th* node of the partition, $C_i(k) = 0$ otherwise, we write the observation equation as

$$\text{(2)} \qquad y(t_k) = C(k)\, \mathbf{u}(t_k) + v(t_k)\,.$$

3. Filtering algorithm. By augmenting the state of the Lumped System with the unknown constant parameter $\boldsymbol{\alpha}$, we obtain the extended system

$$\text{(3)} \qquad \begin{cases} E\dot{\mathbf{u}}(t) &= A(\boldsymbol{\alpha})\mathbf{u}(t) + b\, w(t) \\ \dot{\boldsymbol{\alpha}}(t) &= 0 \end{cases}$$

Let us assume the sampling time to be constant: $t_k - t_{k-1} = T_s$, and, moreover, that the input is applied through a zero-order holder: under these assumptions, we can use a discrete-time EKF [6] to estimate the state of system (3).

Suppose that an estimate of the state and the parameter is available at time t_{k-1}, based on the observations $\{y_1 \ldots y_{k-1}\}$: let us refer to such estimates as $\hat{\mathbf{u}}_{k-1,k-1}$ and $\hat{\boldsymbol{\alpha}}_{k-1,k-1}$. By letting $\Phi_\alpha(t,\tau) = exp\{E^{-1}A(\boldsymbol{\alpha})(t-\tau)\}$, the expression of the predictors at time t_k are (the argument t_k will be replaced by the index k for ease of notation)

$$\begin{cases} \hat{\mathbf{u}}_{k,k-1} &= \Phi_{\hat{\alpha}_{k-1,k-1}}(t_k, t_{k-1})\hat{\mathbf{u}}_{k-1,k-1} + \displaystyle\int_{t_{k-1}}^{t_k} \Phi_{\hat{\alpha}_{k-1,k-1}}(t_k,\sigma)\, b\, w(\sigma)\, d\sigma \\ \hat{\boldsymbol{\alpha}}_{k,k-1} &= \hat{\boldsymbol{\alpha}}_{k-1,k-1} \end{cases}$$

As observation y_k is available, we compute the filtered estimates $\hat{\mathbf{u}}_{k,k}$ and $\hat{\boldsymbol{\alpha}}_{k,k}$ by

$$\text{(4)} \qquad \begin{bmatrix} \hat{\mathbf{u}}_{k,k} \\ \hat{\boldsymbol{\alpha}}_{k,k} \end{bmatrix} = \begin{bmatrix} \hat{\mathbf{u}}_{k,k-1} \\ \hat{\boldsymbol{\alpha}}_{k,k-1} \end{bmatrix} + K_k \left(y_k - C_k \begin{bmatrix} \hat{\mathbf{u}}_{k,k-1} \\ \hat{\boldsymbol{\alpha}}_{k,k-1} \end{bmatrix} \right),$$

where $\mathcal{C}_k = [\,C_k\ \mathbf{0}\,]$, and the Kalman gain $\mathcal{K}_k$ is computed by linearising system (3) about the predictors $\hat{\mathbf{u}}_{k,k-1}$, $\hat{\boldsymbol{\alpha}}_{k,k-1}$; its expression is given recursively by

$$
\begin{aligned}
\mathcal{S}_k &= \mathcal{A}_k \mathcal{P}_{k-1} \mathcal{A}_k' \\
\mathcal{K}_k &= \frac{\mathcal{S}_k \mathcal{C}_k'}{\sigma_V^2 + \mathcal{C}_k \mathcal{S}_k \mathcal{C}_k'} \\
\mathcal{P}_k &= \mathcal{S}_k - \frac{\mathcal{S}_k \mathcal{C}_k' \mathcal{C}_k \mathcal{S}_k}{\sigma_V^2 + \mathcal{C}_k \mathcal{S}_k \mathcal{C}_k'}\,;
\end{aligned}
\tag{5}
$$

matrix $\mathcal{A}_k$ is computed according to

$$
\mathcal{A}_k = \exp\left(T_s \begin{bmatrix} \frac{\partial E^{-1}A(\boldsymbol{\alpha})\mathbf{u}}{\partial \mathbf{u}} & \frac{\partial E^{-1}A(\boldsymbol{\alpha})\mathbf{u}}{\partial \boldsymbol{\alpha}} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} \right),
$$

evaluated at $\mathbf{u} = \hat{\mathbf{u}}_{k,k-1}$ and $\boldsymbol{\alpha} = \hat{\boldsymbol{\alpha}}_{k,k-1}$; bold zeros are null matrices of appropriate dimensions. Matrix $\mathcal{P}_k$ may be regarded as an approximation of the covariance matrix of the estimation error for the extended state: in a standard filtering problem the trace of $\mathcal{P}_k$ is an index of the "goodness" of the estimate at time t_k; by using a moving sensor, it can be minimised, given $\mathcal{P}_{k-1}$, with respect to $\mathcal{C}_k$, which depends only on the sensor position at time t_k. Such a criterion, in a standard filtering problem, has been fruitfully applied [4]; in the present context the meaning of the minimisation of the trace of $\mathcal{P}_k$ is dubious, because $\mathcal{P}_k$ is not properly a covariance matrix. Nevertheless, we can look at this criterion as a heuristic method to improve the properties of the EKF: we propose to choose $x^m(t_k)$ by the rule

$$
x^m(t_k) = \arg \min_{x^m \in \aleph} \{\, tr(\mathcal{P}_k - \mathcal{S}_k) \,\}\,. \tag{6}
$$

The filter we propose is then nothing but an EKF including the step-by-step choice of the sensor position based on the above criterion. We have no proof of convergence or boundedness for that filter but, as we will see in next section, the numerical results are surprisingly good.

In its implementation, a well known device is used: a fictitious input noise covariance matrix $\mathcal{W}$ is added to $\mathcal{S}_k$ in equation (5), so $\mathcal{S}_k$ will given by $\mathcal{S}_k = \mathcal{A}_k \mathcal{P}_{k-1} \mathcal{A}_k' + \mathcal{W}$; this is a classical method to improve the convergence of the filter [7]. Moreover in those cases in which, at some step k, any entry of $\hat{\boldsymbol{\alpha}}_{k,k}$ would result negative, we will take the absolute value.

4. Numerical Results. The results of four numerical experiments are reported: in all of them the true value of the parameter is $a(x) = 10\,exp(x)$; the approximated initial guess $\hat{\boldsymbol{\alpha}}_0$ is assumed to be constant and equal to 10. We consider an approximate model of order $n = 24$, to which a vector parameter $\boldsymbol{\alpha}$ having 25 entries there corresponds; $\mathcal{P}_0$ is $5 \cdot 10^3 I$ and $\mathcal{W}$ is $10^{-2} I$. The output data $\{\, y_k \,\}$ are obtained from a model of order $n = 49$, in order to simulate the continuous-in-space model. The final time for all the simulations is $T_f = 5\,sec.$ and the sampling time is $T_s = 0.01\,sec.$; we define the steady-state state estimation error, in percent, as

$$
\varepsilon_s = 100\, \frac{\sum_1^n |\hat{\mathbf{u}}_i(T_f) - \mathbf{u}_i(T_f)|}{\sum_1^n |\mathbf{u}_i(T_f)|}\,.
$$

In Table 1, the results of four experiments with three different filters are reported. The first filter, called "Moving", is that described in the previous section. The second, denoted by "Fixed", is a standard EKF obtained by fixing the sensor on a node of the partition; we repeat four times that experiment, by fixing the sensor at abscissa $x^m \in \{0.2, 0.4, 0.6, 0.8\}$; the results are relative to the node where "Fixed" works better. The last one ("Filter") is standard Kalman Filter, in which the parameter is fixed at the wrong value $\hat{\alpha}_0$, and which incorporates the same step-by-step choice of the output matrix as "Moving".

Each entry of the table is a pair of numbers, the first of which represents the percentage of successful runs of the corresponding algorithm over a set of twenty realisation of the output noise, and the second is the mean of ε_s over the set of runs in which that filter converges.

In Experiment 1 and Experiment 2 the input distribution function is $f(x) = \delta(x-0.20)$; in Experiment 3 and Experiment 4 we take $f(x) = step(x-0.60) - step(x-0.80)$. Moreover in Experiment 1 and Experiment 3 we assume an output noise with variance $\sigma_V^2 = 10^{-2}$, in Experiment 2 and Experiment 4 we assume $\sigma_V^2 = 10^{-4}$.

5. Conclusions. The numerical experiments show that, by using the proposed filter ("Moving"), the reconstruction of the state is excellent; convergence is obtained in all cases we have tested: this is paid by a greater computational burden with respect to a standard Kalman Filter ("Filter"). Compared with Fixed, the proposed filter seems to work much better, with practically the same computational effort.

In conclusion, the filter we propose is suitable for on-line implementation in those cases in which at least a rough reconstruction of the spatial parameter is essential for a good reconstruction of the state.

	Experiment 1	Experiment 2	Experiment 3	Experiment 4
Moving	100% - 4.77%	100% - 0.52%	100% - 0.94%	100% - 0.28%
Fixed x^m	10% - 98.3% 0.80	30% - 59.3% 0.60	0 - —	0 - —
Filter	100% - 8.59%	100% - 5.61%	100% - 23.8%	100% - 14.9%

Table 1. *Results of the numerical experiments.*

REFERENCES

[1] E. Rafajlowicz. Reduction of distributed system identification complexity using intelligent sensor. *Int. J. Control*, 50(5):1571–1576, 1989.

[2] L. Carotenuto, P. Muraca and G. Raiconi. Optimal location of a moving sensor for the estimation of a distributed-parameter process. *Int. J. Control*, 46(5):1671–1688, 1987.

[3] L. Ljung. Asymptotic behavior of the Extended Kalman Filter as a parameter estimator for linear systems. *IEEE Trans. Automatic Contr.*, AC-24(1):36–50, 1979.

[4] L. Carotenuto, P. Muraca, P. Pugliese and G. Raiconi. Recursive sensor allocation for a class of distributed parameter systems. *Int. J. Modelling and Simulation*, 12(4):117–124, 1992.

[5] G. Strang and G. Fix. *An analisys of the Finite Elements Method.* Prentice-Hall, Inc, 1973.

[6] A. H. Jazwinski. *Stochastic Processes and Filtering Theory.* Academic Press, 1970.

[7] A. H. Jazwinski. Adaptive Filtering. *Automatica*, 5(4):475–485, 1969.