Distributed Finite Element Kalman Filter

G. Battistelli, L. Chisci, N. Forti, G. Pelosi, S. Selleri

Dipartimento di Ingegneria dell'Informazione

Università degli Studi di Firenze

Via Santa Marta 3, 50139 Firenze, Italy

emails: {giorgio.battistelli, luigi.chisci, nicola.forti, giuseppe.pelosi, stefano.selleri}@unifi.it

Abstract— This paper addresses state estimation for spatially distributed systems governed by linear partial differential equations from discrete in-space-and-time noisy measurements provided by sensors deployed over the spatial domain of interest. A decentralised and scalable approach is undertaken by decomposing the domain into overlapping subdomains assigned to different processing nodes interconnected to form a network. Each node runs a local finite-dimensional Kalman filter which exploits the finite element approach for spatial discretisation and the parallel Schwarz method to iteratively enforce consensus on the estimates and covariances over the boundaries of adjacent subdomains. The effectiveness of the proposed distributed consensus-based finite element Kalman filter is demonstrated via simulation experiments concerning a temperature estimation problem modelled by the bi-dimensional heat equation.

Keywords: networked state estimation; distributed-parameter systems; finite element method; Kalman filtering; consensus.

I. INTRODUCTION

Recent advances in wireless sensor network technology make possible to perform cost-effective monitoring of spatially distributed systems by deploying multiple sensors over the area to be monitored. This clearly has several important practical applications including, e.g., weather forecasting, water flow and fire monitoring, monitoring of pollutants. The problem of fusing data from different sensors can be accomplished either in a *centralised* way, i.e. when there is a single fusion center collecting data from all sensors and taking care of the overall spatial domain of interest, or in *distributed* (decentralised) fashion with multiple intercommunicating fusion centres (nodes) each of which can only access part of the sensor data and take care of a smaller region of the overall domain. The decentralised approach is clearly preferable in terms of scalability of computation with respect to the problem size and will be, therefore, pursued in this paper.

Since spatially distributed processes are typically modelled by infinite-dimensional systems, described by *partial differential equations* (PDEs), distributed state estimation for such systems represents a key issue to be addressed. While great attention has been devoted to distributed consensus-type filters for finite-dimensional, both linear and nonlinear, systems [1]-[5] and for multitarget tracking [6], considerably less work is available on the more difficult case of distributed-parameter systems. Important contributions on the topic can

be found in the work of Demetriou [7]-[8] which presents consensus filters for distributed-parameter systems wherein each node of the network aims to estimate the system state on the whole spatial domain of interest.

In the present paper, a different framework is considered wherein each node is only responsible for estimating the state over a smaller region in the neighbourhood of the node. This setup allows for a solution which is *scalable* with respect to the spatial domain (i.e., the computational complexity in each node does not depend on the size of the whole spatial domain but only of its region of competence). In this context, the contribution of the present paper is essentially in two directions. First, we develop *scalable* consensus filters by suitably adapting the so called Schwarz domain decomposition methods [9]-[14], originally conceived to solve a boundary value problem by splitting it into smaller subproblems on subdomains and iterating to achieve consensus among the solutions on adjacent subdomains. Secondly, we exploit the *finite element* (FE) method [15]-[17] in order to approximate the original infinite-dimensional filtering problem into a, possibly large-scale, finite-dimensional one. Combining these two ingredients, we propose a novel distributed *finite element Kalman filter* which generalises to the more challenging distributed case previous work on FE Kalman filtering [18].

The paper is organised as follows. Section II introduces the basic notation and problem formulation. Then Section III presents the centralised FE Kalman filter and subsequently section IV shows how to extend such a filter to the distributed setting by means of parallel Schwarz consensus. Section V demonstrates the effectiveness of the proposed distributed FE Kalman filter via numerical examples related to the estimation of a bi-dimensional temperature field. Finally, section VI ends the paper with concluding remarks.

II. PROBLEM FORMULATION

This paper addresses the estimation of a scalar, time-and-space-dependent, field of interest from given discrete, in both time and space, measurements related to such a field provided by multiple sensors placed within the domain of interest. The scalar field to be estimated $x(\mathbf{p}, t)$ is defined over the space-time domain $\Omega \times \mathbb{R}_+$, as the solution of a *partial differential equation (PDE)* of the form

$$\frac{\partial x}{\partial t} + \mathcal{A}(x) = f \qquad (1)$$

with (possibly unknown) initial condition $x(\mathbf{p}, 0) = x_0(\mathbf{p})$, $\mathbf{p} \in \Omega$, and homogeneous boundary conditions

$$\mathcal{B}(x) = 0 \text{ on } \partial\Omega. \tag{2}$$

The space domain Ω is supposed to be bounded and with smooth boundary $\partial\Omega$.

The measurements

$$y_{q,i} = h_i(x(\mathbf{s}_i, t_q)) + v_{q,i} \tag{3}$$

are provided by sensors $i \in \mathcal{S} \triangleq \{1, \dots, S\}$, located at positions $\mathbf{s}_i \in \Omega$, at discrete sampling instants t_q, $q \in \mathbb{Z}_+ = \{1, 2, \dots\}$, such that $0 < t_1 < t_2 < \cdots$. In (1)-(3): $\mathbf{p} \in \Omega$ denotes the d-dimensional ($d \in \{1, 2, 3\}$) position vector; $\mathcal{A}(\cdot)$ and $\mathcal{B}(\cdot)$ are linear operators over a suitable Hilbert space V, with $\mathcal{A}(\cdot)$ self-adjoint and coercive; $f(\mathbf{p}, t)$ is a forcing term possibly affected by process noise; $h_i(\cdot)$ is the measurement function of sensor i; $v_{q,1}, \dots, v_{q,N}$ are mutually independent white measurement noise sequences, also independent from the initial state $x_0(\mathbf{p}) = x(\mathbf{p}, 0)$ for any $\mathbf{p} \in \Omega$.

More precisely, the aim is to estimate $x(\mathbf{p}, t)$ given the information set $Y^t \triangleq \{y_{q,i}, \forall i \in \mathcal{S} \text{ and } \forall q : t_q \leq t\}$. This is clearly an infinite-dimensional filtering problem. In the next section, it will be shown how it can be approximated into a finite-dimensional filtering problem by exploiting the FE method [15]-[16].

An example of the above general problem is the estimation of the temperature field x over the spatial domain of interest given point measurements of temperature sensors. In this case, V is usually taken as the Sobolev space $H^1(\Omega)$, the measurement function is simply $h(x) = x$, while the PDE (1) reduces to the well known *heat equation* with $\mathcal{A}(x) = -\nabla \cdot (\lambda \nabla(x))$ and $\mathcal{B}(x) = \alpha \, \partial x / \partial \mathbf{n} + \beta x$ with $\alpha(\mathbf{p})\beta(\mathbf{p}) \geq 0$, $\alpha(\mathbf{p}) + \beta(\mathbf{p}) > 0$, $\forall \mathbf{p} \in \partial\Omega$. Here $\lambda(\mathbf{p})$ is the *thermal diffusivity*, $\cdot$ stands for scalar product, $\nabla \triangleq \partial / \partial \mathbf{p}$ denotes the gradient operator, $\mathbf{n}$ is the outward pointing unit normal vector of the boundary $\partial\Omega$, and $\partial x / \partial \mathbf{n} = \nabla x \cdot \mathbf{n}$. Clearly, when the thermal diffusivity is space-independent, one has $\mathcal{A}(x) = -\lambda \nabla^2(x)$, where $\nabla^2 = \nabla \cdot \nabla$ is the Laplacian operator.

Notice that considering homogeneous boundary conditions as in (2) is not restrictive, since the non-homogeneous case $\mathcal{B}(x) = g$ on $\partial\Omega$ can be subsumed into the homogeneous one by means of the change of variables $z = x - w$, where w is any function belonging to V and satisfying the non-homogeneous boundary conditions.

III. CENTRALISED FINITE ELEMENT KALMAN FILTER

In this section, it is shown how to approximate the continuous-time infinite-dimensional system (1) into a discrete-time finite-dimensional linear dynamical system within the FE framework.

By subdividing the domain Ω into a suitable set of non overlapping regions, or elements, and by defining a suitable set of basis functions $\phi_j(\mathbf{p}) \in V$ ($j = 1, \dots, n$) on them, it is possible to write an approximation of the unknown function $x(\mathbf{p}, t)$ as

$$x(\mathbf{p}, t) = \sum_{j=1}^{n} \phi_j(\mathbf{p}) \, x_j(t) = \boldsymbol{\phi}^T(\mathbf{p}) \, \mathbf{x}(t) \tag{4}$$

where: $x_j(t)$ is the unknown expansion coefficient of function $x(\mathbf{p}, t)$ relative to time t and basis function $\phi_j(\mathbf{p})$; $\boldsymbol{\phi}(\mathbf{p}) \triangleq col\{\phi_j(\mathbf{p})\}_{j=1}^{n}$ and $\mathbf{x}(t) \triangleq col\{x_j(t)\}_{j=1}^{n}$.

The choices of the basis functions ϕ_j and of the elements are key points of the FE method. Typically, the elements (triangles or quadrilaterals in 2D, polyhedral in 3D) define a FE mesh with vertices $\mathbf{p}_j \in \Omega, j = 1, \dots, n$. Then each basis function ϕ_j is a piece-wise polynomial which vanishes outside the FEs around $\mathbf{p}_j$ and such that $\phi_j(\mathbf{p}_i) = \delta_{ij}$, δ_{ij} denoting the Kronecker delta.

Exploiting (4) in (1), applying the Galerkin weighted residual method, and thanks to the linearity of operator $\mathcal{A}(\cdot)$ the usual FE weak form is obtained [15]-[16]

$$\underbrace{\left[\int_\Omega \boldsymbol{\phi}(\mathbf{p})\boldsymbol{\phi}^T(\mathbf{p})d\mathbf{p} \right]}_{\mathbf{M}} \dot{\mathbf{x}}(t) + \underbrace{\left[\int_\Omega \boldsymbol{\phi}(\mathbf{p}) \left[\mathcal{A}(\boldsymbol{\phi}(\mathbf{p})) \right]^T d\mathbf{p} \right]}_{-\mathbf{S}} \mathbf{x}(t)$$
$$= \underbrace{\int_\Omega \boldsymbol{\phi}(\mathbf{p})f(\mathbf{p}, t)d\mathbf{p}}_{\mathbf{u}(t)}$$

$$\tag{5}$$

where $\mathcal{A}(\boldsymbol{\phi}) \triangleq col\{\mathcal{A}(\phi_j)\}_{j=1}^{n}$. It is evident how the first two integrals in (5) depend only on basis functions and can be computed *a priori*. In particular, the first integral yields the well known *mass* matrix $\mathbf{M}$, while the second depends on the operator $\mathcal{A}(\cdot)$ and, in the thermal case, is the stiffness matrix $\mathbf{S}$ [15]. The third integral depends on the forcing term f, which is assumed to be known, and can hence be computed *a priori*, leading to a time dependent vector contribution $\mathbf{u}(t)$. The resulting linear differential equation $\mathbf{M}\dot{\mathbf{x}} = \mathbf{S}\mathbf{x} + \mathbf{u}$, where $\mathbf{M}$ turns out to be invertible by linear independence of the basis functions $\phi_j(\cdot)$, can be discretised in time by different methods (e.g., backward or forward Euler integration, or the zero-order-hold method) to provide the discrete-time state-space model

$$\mathbf{x}_{k+1} = \mathbf{A}\mathbf{x}_k + \mathbf{B}\mathbf{u}_k + \mathbf{w}_k \tag{6}$$

where the process noise $\mathbf{w}_k$ has been introduced to account for the various uncertainties and/or imprecisions (e.g. FE approximation, time discretisation, and imprecise knowledge of boundary conditions). Specifically, the backward Euler method (here adopted for stability issues) leads to a marching in time FE implementation [17] which yields (6) with

$$\mathbf{A} = \left(\mathbf{I} - \Delta \mathbf{M}^{-1}\mathbf{S} \right)^{-1}, \quad \mathbf{B} = \mathbf{A}\mathbf{M}^{-1}\Delta,$$
$$\mathbf{u}_k \triangleq \mathbf{u}((k+1)\Delta), \mathbf{x}_k \triangleq \mathbf{x}(k\Delta) = col\{x_j(k\Delta)\}_{j=1}^{n}$$

where Δ denotes the time integration interval.

In the following, for the sake of notational simplicity, it will be assumed that each sampling instant is a multiple of Δ, i.e., $t_q = T_q\Delta$ with $T_q \in \mathbb{Z}_+$, and we let $\mathcal{T} = \{T_1, T_2, \dots\}$;

irregular sampling could, however, be easily dealt with. This amounts to assuming that the numerical integration rate of the PDE (1) in the filter can be higher than the measurement collection rate, which can be useful in order to reduce the numerical errors. In a centralised setting where all sensor measurements are available to the filter, the measurement equation (3) takes the discrete-time form

$$\mathbf{y}_k = \mathbf{h}(\mathbf{x}_k) + \mathbf{v}_k \qquad (7)$$

for any $k = T_q \in \mathcal{T}$, where

$$\mathbf{y}_k \triangleq col\{y_{q,i}\}_{i \in \mathcal{S}}, \ \mathbf{h}(\mathbf{x}) \triangleq col\left\{ h_i\left(\boldsymbol{\phi}^T(\mathbf{s}_i)\mathbf{x}\right)\right\}_{i \in \mathcal{S}},$$
$$\mathbf{v}_k \triangleq col\{v_{q,i}\}_{i \in \mathcal{S}}$$

In particular, in the case wherein all sensors directly measure the target field x, i.e. $h_i(x) = x$ for all $i \in \mathcal{S}$, the measurement equation (7) turns out to be linear with $\mathbf{h}(\mathbf{x}) = \mathbf{Cx}$, where

$$\mathbf{C} = col\left\{\boldsymbol{\phi}^T(\mathbf{s}_i)\right\}_{i \in \mathcal{S}} \qquad (8)$$

Summarizing, the original infinite-dimensional continuous-time problem has been reduced to a much simpler finite-dimensional (possibly large-scale) discrete time filtering problem (a linear one provided that all sensor measurement functions are linear) to which the *Kalman filter*, or *extended Kalman filter* when sensor nonlinearities are considered, can be readily applied. The resulting centralised filter recursion becomes:

$$\hat{\mathbf{x}}_{k|k} = \begin{cases} \hat{\mathbf{x}}_{k|k-1} + \mathbf{L}_k\left(\mathbf{y}_k - \mathbf{h}\left(\hat{\mathbf{x}}_{k|k-1}\right)\right) & \text{if } k \in \mathcal{T} \\ \hat{\mathbf{x}}_{k|k-1} & \text{otherwise} \end{cases}$$
$$\mathbf{P}_{k|k} = \begin{cases} \mathbf{P}_{k|k-1} - \mathbf{L}_k\mathbf{C}_k^T\mathbf{P}_{k|k-1} & \text{if } k \in \mathcal{T} \\ \mathbf{P}_{k|k-1} & \text{otherwise} \end{cases}$$
$$\hat{\mathbf{x}}_{k+1|k} = \mathbf{A}\hat{\mathbf{x}}_{k|k} + \mathbf{Bu}_k$$
$$\mathbf{P}_{k+1|k} = \mathbf{AP}_{k|k}\mathbf{A}^T + \mathbf{Q}_k \qquad (9)$$

where

$$\mathbf{C}_k = \frac{\partial \mathbf{h}}{\partial \mathbf{x}}\left(\hat{\mathbf{x}}_{k|k-1}\right)$$
$$\mathbf{L}_k = \mathbf{P}_{k|k-1}\mathbf{C}_k\left(\mathbf{R}_k + \mathbf{C}_k\mathbf{P}_{k|k-1}\mathbf{C}_k^T\right)^{-1}$$

for $k \in \mathcal{T}$. The recursion is initialized from suitable $\hat{\mathbf{x}}_{1|0}$ and $\mathbf{P}_{1|0} = \mathbf{P}_{1|0}^T > \mathbf{0}$. In (9), $\mathbf{Q}_k$ and $\mathbf{R}_k$ denote the covariance matrices of the process noise $\mathbf{w}_k$ and, respectively, measurement noise $\mathbf{v}_k$, which are assumed as usual to be white, zero-mean, mutually uncorrelated and also uncorrelated with the initial state $\mathbf{x}_1$.

IV. DISTRIBUTED FINITE ELEMENT KALMAN FILTER

In order to develop a scalable distributed filter for monitoring the target field, the idea is to decompose the original problem on the whole domain of interest into estimation subproblems concerning smaller subdomains, and then to assign such subproblems to different *nodes* which can locally process and exchange data. To this end, let us consider the set of nodes $\mathcal{N} = \{1, \ldots, N\}$, subdivide the domain Ω into possibly overlapping subdomains Ω_m, $m \in \mathcal{N}$, such that $\Omega = \bigcup_{m \in \mathcal{N}} \Omega_m$, and assign the task "estimation of x over

Ω_m" to node m. Further, let $\mathbf{y}_q^m \triangleq col\{y_{q,i} : \mathbf{s}_i \in \Omega_m\}$ denote the vector of local measurements available to node m at time t_q.

Hence, the idea is to run in each node $m \in \mathcal{N}$ a field estimator for the region Ω_m exploiting local measurements $\mathbf{y}_q^m$, information from the nodes assigned to neighbouring subdomains, as well as the PDE model (1) properly discretised in time and space. Taking inspiration from the Schwarz method [9]-[11], neighbouring local estimators should iteratively find a *consensus* on the estimates concerning the common parts. The Schwarz method has been originally conceived [9] for an iterative solution of boundary value problems. Subsequently, it has received renewed interest [10]-[11] in connection with the parallelisation of PDE solvers. In loose terms, the idea of the *parallel Schwarz method* is to decompose the original PDE problem on the overall domain of interest into subproblems concerning smaller subdomains, and then to solve in parallel such subproblems via iterations in which previous solutions concerning neighbouring subdomains are used as boundary conditions. As shown below, such an idea turns out to be especially useful for the distributed filtering problem considered in this work.

To formalise the consensus let us define, for any $m \in \mathcal{N}$, a partition $\{\Gamma_{mj}\}_{j \in \mathcal{N}_m}$ of $\partial\Omega_m$ (the boundary of Ω_m) such that

$$\begin{aligned} \Gamma_{mm} &= \partial\Omega \cap \partial\Omega_m \\ \partial\Omega_m &= \bigcup_{j \in \mathcal{N}_m} \Gamma_{mj} \\ \Gamma_{mj} &\subset \Omega_j, \quad \forall j \neq m \\ \Gamma_{mj} \cap \Gamma_{mh} &= \emptyset, \quad \forall j \neq h \end{aligned} \qquad (10)$$

In this way, each piece Γ_{mj} of $\partial\Omega_m$ for any $j \in \mathcal{N}_m\backslash\{m\}$ is uniquely assigned to node j.

Hence, node j is called an in-neighbour of node m whenever $\Gamma_{mj} \neq \emptyset$; this clearly originates a directed network (graph) $\mathcal{G} = (\mathcal{N}, \mathcal{L})$ with node set $\mathcal{N}$ and link set $\mathcal{L} \triangleq \{(j, m) \in \mathcal{N} \times \mathcal{N} : \Gamma_{mj} \neq \emptyset\}$. It is also convenient to define, for each node m, its in-neighbourhood $\mathcal{N}_m \triangleq \{j : (j, m) \in \mathcal{L}\}$ which, by definition, includes node m.

In order to describe the filtering cycle to be implemented in node m within the sampling interval $[t_q, t_{q+1})$, let us assume that at time t_q^-, before the acquisition of $\mathbf{y}_q^m$, such a node is provided with a prior estimate $\hat{\mathbf{x}}_{q|q-1}^m$ and relative covariance $\mathbf{P}_{q|q-1}^m$, as the result of the previous filtering cycles. Let δ be the time interval necessary for performing one consensus step, i.e., information exchange between neighbours and related computations. Then, $L_q \triangleq (t_{q+1} - t_q)/\delta$ represents the number of consensus steps (equal to the number of allowed data exchanges) in the q-th sampling interval. Note that, for the sake of notational simplicity, hereafter it is supposed that $t_{q+1} - t_q$ is an integer multiple of δ, i.e., $L_q \in \mathbb{Z}_+$. Anyway, the method could easily encompass the general case. Then, the above mentioned filtering cycle for the proposed distributed estimation algorithm essentially consists of:

1) **Correction**, i.e. incorporation (assimilation) of the last measurement $\mathbf{y}_q^m$ into the current estimate;
2) **Consensus**, i.e. alternate exchanges of estimates with the neighbouround $\mathcal{N}_m$ and predictions over the time sub-intervals $[t_q + (\ell-1)\delta, t_q + \ell\delta]$ for $\ell = 1, \ldots, L_q$, i.e. L_q times.

The proposed *Parallel Schwarz Consensus* filter is detailed hereafter.

1) Given $\mathbf{y}_q^m$, update the prior estimate $\hat{\mathbf{x}}_{q|q-1}^m$ and covariance $\mathbf{P}_{q|q-1}^m$ into $\hat{\mathbf{x}}_{q|q}^m$ and $\mathbf{P}_{q|q}^m$.
2) Initialize the consensus with $\hat{\mathbf{x}}_{q,0}^m = \hat{\mathbf{x}}_{q|q}^m$ and $\mathbf{P}_{q,0}^m = \mathbf{P}_{q|q}^m$.
3) For $\ell = 1, \ldots, L_q$ proceed as follows
   a) Exchange data with the neighbourhood; specifically send to neighbour j the data $\hat{x}_{q,\ell-1}^m, \mathbf{P}_{q,\ell-1}^m$ concerning the sub-boundary $\Gamma_{jm} \subset \partial\Omega_j$, and get from neighbour j the data $\hat{x}_{q,\ell-1}^j, \mathbf{P}_{q,\ell-1}^j$ concerning the sub-boundary $\Gamma_{mj} \subset \partial\Omega_m$.
   b) Solve the problem

   $$\frac{\hat{x}_{q,\ell}^m - \hat{x}_{q,\ell-1}^m}{\delta} + \mathcal{A}\left(\hat{x}_{q,\ell}^m\right) = f_{q,\ell} \quad \text{in } \Omega_m \quad (11)$$

   subject to the Dirichlet boundary conditions

   $$\hat{x}_{q,\ell}^m = \hat{x}_{q,\ell-1}^j \quad \text{on } \Gamma_{mj} \quad \forall j \in \mathcal{N}_m \backslash \{m\} \quad (12)$$

   and the linear boundary conditions

   $$\mathcal{B}(\hat{x}_{q,\ell}^m) = 0 \text{ on } \Gamma_{mm} . \quad (13)$$

   where $f_{q,\ell}(\mathbf{p}) \triangleq f(\mathbf{p}, t_q + \ell\delta)$.
4) Set $\hat{\mathbf{x}}_{q+1|q}^m = \hat{\mathbf{x}}_{q,L}^m$ and $\mathbf{P}_{q+1|q}^m = \mathbf{P}_{q,L}^m$ for the next cycle.

A number of remarks concerning the above reported algorithm are in order. First of all, notice that the prediction step of each local filter is directly incorporated into the consensus algorithm. Clearly, the boundary value problem (11)-(13) has to be solved via a finite dimensional approximation. In particular, we follow the same approach described in Section III for the centralised case by independently constructing a FE mesh for each subdomain Ω_m, and then applying the Galerkin weighted residual method so as to derive a local discrete-time discrete-space model. Hence, the numerical solution of (11)-(13) takes the form

$$\hat{\mathbf{x}}_{q,\ell}^m = \mathbf{A}^m \hat{\mathbf{x}}_{q,\ell-1}^m + \sum_{j \in \mathcal{N}_m \backslash \{m\}} \mathbf{A}^{mj} \hat{\mathbf{x}}_{q,\ell-1}^j + \mathbf{B}^m \mathbf{u}_{q,\ell-1}^m$$

$$(14)$$

The additional term $\sum_{j \in \mathcal{N}_m \backslash \{m\}} \mathbf{A}^{mj} \hat{\mathbf{x}}_{q,\ell-1}^j$ arises from the non-homogeneous Dirichlet boundary conditions (12). With this respect, it is worth noting that the matrices $\mathbf{A}^{mj}$ are sparse since only the components of the neighbour estimate $\hat{\mathbf{x}}_{q,\ell-1}^j$ concerning the sub-boundary Γ_{mj} are involved. In accordance with (14), the covariance matrix $\mathbf{P}_{q,\ell}^m$ associated to the predicted estimate $\hat{\mathbf{x}}_{q,\ell}^m$ is determined as

$$\begin{aligned} \mathbf{P}_{q,\ell}^m &= \mathbf{A}^m \mathbf{P}_{q,\ell-1}^m \left(\mathbf{A}^m\right)^T \\ &+ \sum_{j \in \mathcal{N}_m \backslash \{m\}} \mathbf{A}^{mj} \mathbf{P}_{q,\ell-1}^j \left(\mathbf{A}^{mj}\right)^T + \mathbf{Q}^m (15) \end{aligned}$$

As in the centralised context, the term $\mathbf{Q}^m$ accounts for the various uncertainties and imprecisions (i.e., discretisation errors, imprecise knowledge of the exogenous input f and of the boundary conditions (13)). The additional term $\sum_{j \in \mathcal{N}_m \backslash \{m\}} \mathbf{A}^{mj} \mathbf{P}_{q,\ell-1}^j \left(\mathbf{A}^{mj}\right)^T$, which instead is peculiar of the decentralised case, accounts for the fact that each neighbour estimate $\hat{\mathbf{x}}_{q,\ell-1}^j$ is affected by an estimation error with estimated covariance equal to $\mathbf{P}_{q,\ell-1}^j$. It is worth noting that equation (15) is based on the idea (already proposed in [3] in the context of distributed estimation of large-scale interconnected systems) of treating each neighbour state as an unknown stochastic input with mean $\hat{\mathbf{x}}_{q,\ell-1}^j$ and covariance $\mathbf{P}_{q,\ell-1}^j$.

As already pointed out, the proposed consensus algorithm is based on the parallel Schwarz method for evolution problems, which, as well known, enjoys nice convergence properties to the centralised solution as the time discretisation step δ tends to zero [10]-[11]. Hence, it seems a sensible and promising approach to spread the information through the network.

Finally, the correction step of the local filtering cycle is the usual (extended) Kalman filter update step for the local subsystem and takes the form

$$\begin{aligned} \hat{\mathbf{x}}_{q|q}^m &= \hat{\mathbf{x}}_{q|q-1}^m + \mathbf{L}_q^m \left(\mathbf{y}_q^m - \mathbf{h}^m \left(\hat{\mathbf{x}}_{q|q-1}^m\right)\right) \\ \mathbf{P}_{q|q}^m &= \mathbf{P}_{q|q-1}^m - \mathbf{L}_q^m (\mathbf{C}_q^m)^T \mathbf{P}_{q|q-1}^m \\ \mathbf{C}_q^m &= \frac{\partial \mathbf{h}^m}{\partial \mathbf{x}} \left(\hat{\mathbf{x}}_{q|q-1}^m\right) \\ \mathbf{L}_q^m &= \mathbf{P}_{q|q-1}^m \mathbf{C}_q^m \left(\mathbf{R}_q^m + \mathbf{C}_q^m \mathbf{P}_{q|q-1}^m (\mathbf{C}_q^m)^T\right)^{-1} \end{aligned}$$

where $\mathbf{h}^m \triangleq col\{h_i : \mathbf{s}_i \in \Omega_m\}$ denote the local measurement function at node m.

V. NUMERICAL EXAMPLES

In this section, we provide some numerical results to illustrate the performance of the proposed distributed *finite element Kalman filter* described in section IV. As a particular example of (1), we consider the transient *heat conduction* problem (see section II) in a thin polygonal metal plate with constant, homogeneous, and isotropic properties. Suppose that the plate is relatively thin compared to the planar dimensions, then it is appropriate to reduce the problem to two dimensions, since the temperature can be assumed to be constant along the thickness direction. Hence, the field $x(\xi, \eta, t)$ to be estimated in a thin plate is modeled by the bi-dimensional parabolic PDE $\partial x/\partial t - \lambda \left(\partial^2 x/\partial \xi^2 + \partial^2 x/\partial \eta^2\right) = f$ with boundary condition $\mathcal{B}(x) = \alpha(\xi, \eta) \partial x/\partial \mathbf{n} + \beta(\xi, \eta) x$ such that $\alpha(\xi, \eta) \beta(\xi, \eta) \geq 0$, $\alpha(\xi, \eta) + \beta(\xi, \eta) > 0$, $\forall (\xi, \eta) \in \partial\Omega$. Here ξ and η are the Cartesian position coordinates, i.e. $\mathbf{p} = [\xi, \eta]^T$, whereas $\lambda = 116.6 \times 10^{-6} \left[m^2/s\right]$ (*thermal diffusivity* of copper at $293 [K]$) is assumed to be constant in time and space.

Experiment 1

In the first example, transient analysis is performed on a thin adiabatic plate with a fixed temperature along the bottom

edge and no inner heat-generation, monitored by $S = 16$ sensor nodes. This is a problem with mixed boundary conditions, namely a non-homogeneous Dirichlet condition on the bottom edge of the plate, and natural homogeneous Neumann boundary conditions on the remaining insulated sides, where $\partial x / \partial \mathbf{n} = 0$ holds. Note that both distributed and centralised filters are unaware of the real system boundary conditions, so they simply assume the plate adiabatic on each side.

As shown in Fig. 1, the domain under consideration is decomposed into $N = 16$ overlapping subdomains Ω_m, i.e. $\mathcal{N} = \{1, \ldots, 16\}$, each being assigned to a *node* with local processing and communication capabilities. Notice that, for simplicity, it has been assumed that $N = S$ and that there is only a single sensor within each subdomain Ω_m. The communication topology which defines neighbouring nodes is shown in Fig. 1. Sensors i located in the known positions $\mathbf{s}_i = [\xi_i, \eta_i]^T$ are assumed to collect point temperature measurements at regularly time-spaced instants $t_q = q T_s$, with $T_s = 100 \, [s]$ and standard deviation of measurement noise $\sigma_v = 0.1 \, [K]$. In order to run a temperature field estimator in each region Ω_m, the matrices in (14), to be used by the local filters, are obtained off-line by FE approximation (see section III) and time-discretisation with fixed sampling interval $\delta = T_s / L$. A triangular mesh (see Fig. 1) of size $b = 0.2$ (defined as the length of the longest edge of the element), is generated for each bounded domain with polygonal boundary. Domain triangulation allows for a simple construction of basis functions $\{\phi_j(\xi, \eta)\}_{j=1}^n$ (4), here chosen as two dimensional hat functions. These are defined on each element as a linear polynomial $\gamma_{\mathcal{E}}(\xi, \eta) = c_0 + c_1 \xi + c_2 \eta$ with $(\xi, \eta) \in \mathcal{E}$ and $c_0, c_1, c_2 \in \mathbb{R}$. Such a function is uniquely determined by its three nodal values $x_i = \gamma_{\mathcal{E}}(\xi_i, \eta_i)$, $i \in \mathcal{E}$. Each hat function ϕ_j such that

$$\phi_j(\xi_i, \eta_i) = \begin{cases} 1 & \text{if } i = j \\ 0 & \text{if } i \neq j \end{cases} \qquad i, j = 1, 2, ..., n$$

is, therefore, a continuous piecewise linear function with a small support defined by the set of triangles sharing node j. Basis functions are used at a later stage for the element-by-element construction, in each *node* m, of matrices $\mathbf{S}_m$ and $\mathbf{M}_m$, which provide the finite-dimensional model of temperature evolution in Ω_m. The initial temperature field of the plate is $293 \, [K]$, and the bottom edge is set to $298 \, [K]$. Furthermore, the a-priori estimate taken as first guess of the field of interest is $296 \, [K]$, with diagonal covariance $\mathbf{P}_{1|0} = 20 \, \mathbf{I}$. A zero-mean white noise process has been assumed, with covariance $\mathbf{Q} = \sigma_w^2 \mathbf{I}$, where $\sigma_w = 3 \, [K]$.

The performance of the novel distributed *FE Kalman filter* has been evaluated in terms of Root Mean Square Error (RMSE) of the estimated temperature field, averaged over a set of sampling points evenly spread within Ω, and 100 Monte Carlo trials. The duration of each simulation experiment is fixed to $2 \times 10^4 \, [s]$ (200 samples). The *ground truth* of the experiments is represented by a real process simulator implementing a finer mesh ($b = 0.1$ instead of $b = 0.2$), and aware of the time-varying boundary conditions

of the system. Fig. 2 illustrates the performance comparison between centralised (cFE-KF) and distributed (dFE-KF) filters for different values of the parameter L (number of consensus iterations) adopted in the distributed framework. It can be seen that the performance of the distributed FE filters is very close, even for $L = 1$, to that of the centralised filter, which collects all the data in a central node. Further, the RMSE behaviour improves by increasing the number of consensus steps.
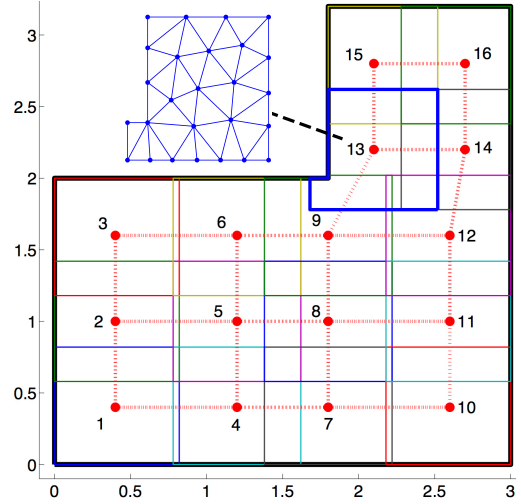

Fig. 1: Domain decomposition into 16 subdomains (solid rectangles) and graph structure (red dotted lines) of the dFE-KF, with a detail of the mesh approximating Ω_{13}. The position of each sensor is denoted by a circle.
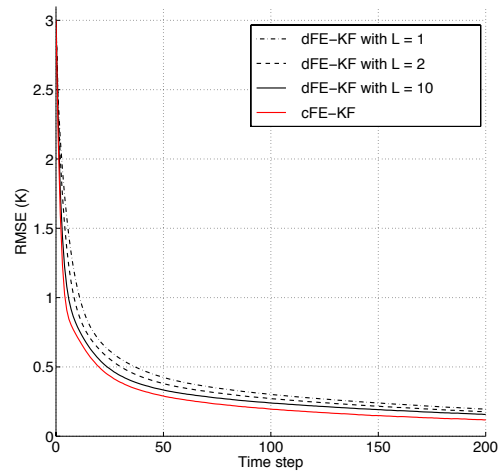

Fig. 2: RMSE of the FE filters in the first experiment.

Experiment 2

In the second simulation example, different boundary conditions are considered. Specifically, a time-dependent Dirichlet condition of $293 \, [K]$ for time steps $q \in \{0, ..., 49\}$, and $301 \, [K]$ for $q \in \{50, ..., 350\}$, is set on all nodes of the bottom edge. The right edge of the plate is first assumed adiabatic for $q \in \{0, ..., 119\}$, then a Dirichlet
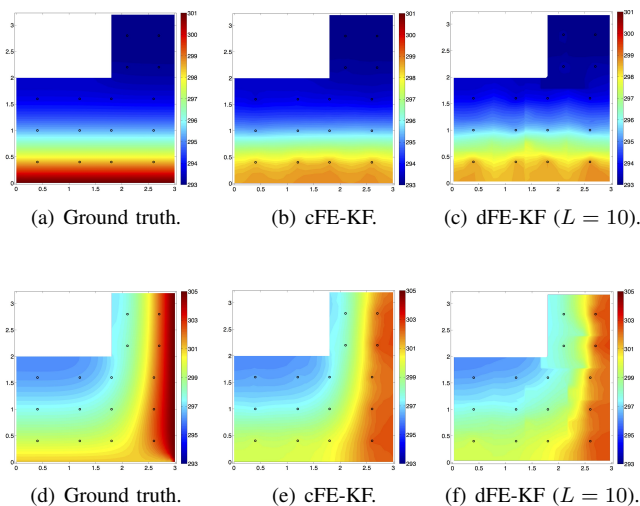
(a) Ground truth. (b) cFE-KF. (c) dFE-KF ($L = 10$).

(d) Ground truth. (e) cFE-KF. (f) dFE-KF ($L = 10$).

Fig. 3: *True* and estimated temperature fields in experiment 2 at time steps $q = 88$ (a,b,c) and $q = 150$ (d,e,f).

Fig. 4: RMSE of the FE filters in the second experiment.

boundary condition is applied for $q \in \{120, ..., 350\}$. The remaining edges are assumed thermally insulated for the duration of the whole experiment. In addition, we consider the initial temperature field on the plate uniform at $293\,[K]$. As previously assumed in experiment 1, the centralised and distributed filters have no knowledge of boundary conditions. Performance of the proposed distributed filter has been evaluated for different values of L over 100 independent Monte Carlo runs and compared to the behaviour of the centralised FE Kalman filter. Simulation results, in Fig. 4, show that the proposed FE estimators provide comparable performance to the centralised filter, moreover the gap reduces as L increases. It is worth pointing out that the spikes appearing in the RMSE plot, displayed in Fig. 4, are due to the rapid change of the unknown boundary conditions, which cause a considerable increase of the estimation error. Nevertheless, the filters under consideration manage to compensate for the lack of knowledge and effectively reduce the error, even if, due to these persistent disturbances, errors do not converge to zero.

VI. CONCLUSIONS

The paper has dealt with the decentralised estimation of a time-evolving and space-dependent field governed by a linear partial differential equation, given point-in space measurements of multiple sensors deployed over the area of interest. The originally infinite-dimensional filtering problem has been approximated into a finite-dimensional large-scale one via the *finite element* method and, further, a consensus approach inspired by the parallel Schwarz method for domain decomposition has allowed to nicely scale the overall problem complexity with respect to the number of used processing nodes. Combining these two ingredients, a novel computationally efficient consensus finite-element Kalman filter has been proposed to solve in a decentralised and scalable fashion filtering problems involving distributed-parameter systems.
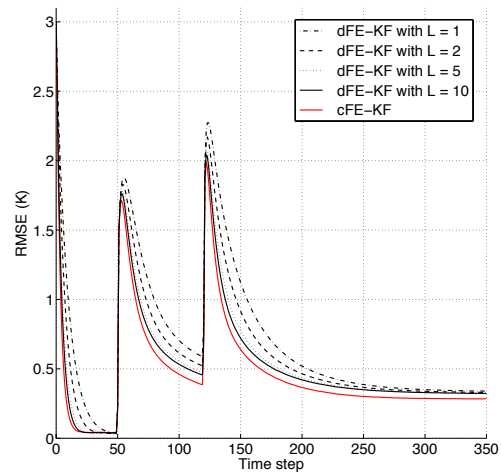
REFERENCES

[1] U. A. Khan and J. M. F. Moura, "Distributing the Kalman filter for large-scale systems", *IEEE Trans. on Signal Processing*, vol. 66, pp. 4919–4935, 2008.
[2] S. Stankovic, M. S. Stankovic, and D. M. Stepanovic, "Consensus based overlapping decentralized estimation with missing observations and communication faults", *Automatica*, vol. 45, pp. 1397–1406, 2009.
[3] M. Farina, G. Ferrari-Trecate, and R. Scattolini, "Moving-horizon partition-based state estimation of large-scale systems", *Automatica*, vol. 46, pp. 910–918, 2010.
[4] G. Battistelli and L. Chisci, "Kullback-Leibler average, consensus on probability densities, and distributed state estimation with guaranteed stability", *Automatica*, vol. 5, pp. 707-718, 2014.
[5] G. Battistelli, L. Chisci, and C. Fantacci, "Parallel consensus on likelihoods and priors for networked nonlinear filtering", *IEEE Signal Processing Letters*, vol. 21, pp. 787-791, 2014.
[6] G. Battistelli, L. Chisci, C. Fantacci, A. Farina, and A. Graziano, "Consensus CPHD filter for distributed multitarget tracking", *IEEE Journal of Selected Topics in Signal Processing*, vol. 7, pp. 508-520, 2013.
[7] M. A. Demetriou, "Design of consensus and adaptive consensus filters for distributed parameter systems", *Automatica*, vol. 46, pp. 300–311, 2010.
[8] M. A. Demetriou, "Adaptive consensus filters of spatially distributed systems with limited connectivity", *Proc, 52nd IEEE Conference on Decision and Control*, pp. 442-447, Firenze, Italy, 2013.
[9] H. A. Schwarz, *Gesammelte mathematische Abhandlungen. Band I, II*, AMS Chelsea Publishing, Bronx, NY, 1890.
[10] P. L. Lions, *On the Schwarz alternating method. I*, in First Int. Symp. on Domain Decomposition Methods for Partial Differential Equations, R. Glowinski et al., SIAM, Philadelphia, PA, pp. 1–42, 1988,.
[11] M. J. Gander, "Schwarz methods over the course of time", *Electronic Trans. on Numerical Analysis*, vol. 31, pp. 228–255, 2008.
[12] T. F. Chan and T. P. Mathew, "Domain decomposition algorithms", *Acta Numerica*, vol. 3, pp. 61–143, 1994.
[13] A. Toselli and O. Widlund, *Domain decomposition methods – Algorithms and theory*, Springer–Verlag, Berlin, Germany, 2005.
[14] B. Smith, P. Bjorstad, and W. Gropp, *Domain decomposition: parallel multilevel methods for elliptic partial differential equations*, Cambridge University Press, New York, NY, 1996.
[15] G. Pelosi, R. Coccioli, and S. Selleri, *Quick finite elements for electromagnetic waves*, Artech House, Norwood, MA, 2009.
[16] S. C. Brenner and L. R. Scott, *The mathematical theory of finite element methods*, Springer–Verlag, New York, NY, 1996.
[17] J.-F. Lee, J.-F. and Z. Sacks, "Whitney elements time domain (WETD) methods", *IEEE Trans. on Magnetics.*, vol. 31, pp. 1325-1329, 1995.
[18] R. Suga and M. Kawahara, "Estimation of tidal current using Kalman filter finite-element method", *Computers & Mathematics with Applications*, vol. 52, pp. 1289-1298, 2006.