This is a preprint of a paper that appeared in
Procedia Computer Science (ICCS 2017), Vol. 108, Elsevier, 2017

Decentralized Dynamic Data-Driven Monitoring of Dispersion Processes on Partitioned Domains

Tobias Ritter[1,2]*, Stefan Ulbrich[1,3], and Oskar von Stryk[1,2]

[1] Graduate School Computational Engineering
[2] Department of Computer Science
{ritter,stryk}@sim.tu-darmstadt.de
[3] Department of Mathematics
ulbrich@mathematik.tu-darmstadt.de
Technische Universität Darmstadt, Germany

Abstract
The application of mobile sensor-carrying vehicles for online estimating dynamic dispersion processes is extremely beneficial. Based on current estimates that depend on past measurements and forecasts obtained from a discretized PDE-model, the movement of the vehicles can be adapted resulting in measurements at more informative locations. In this work, a novel decentralized monitoring approach based on a partitioning of the spatial domain into several subdomains is proposed. Each sensor is assigned to the subdomain it is located in and is only required to maintain a process and multi-vehicle model related to its subdomain. In this way, vast communication requirements of related centralized approaches and costly full model simulations are avoided making the presented approach more scalable with respect to a larger number of sensor-carrying vehicles and a larger problem domain. The approach consists of a new prediction and update method based on a domain decomposition method and a partitioned variant of the Ensemble Square Root Filter getting along with a minimum exchange of data between sensors on neighboring subdomains. Furthermore, a cooperative vehicle controller is applied in such a way that a dynamic adaption of the sensor distribution becomes possible.

Keywords: Dynamic data-driven application system, Domain decomposition, State estimation, Ensemble Square Root Filter, Decentralized estimation, Vehicle control

1 Introduction

Estimating atmospheric or underwater pollutant dispersion, predicting the impacts of severe weather conditions, detecting wildfires - all these applications require the monitoring of dynamical spatially distributed dispersion processes. A typical goal of the monitoring process is the generation of a repeatedly updated online estimate of the current process state. For this

*The work of Tobias Ritter has been partially supported by the 'Excellence Initiative' of the German Federal and State Governments and the Graduate School of Computational Engineering at TU Darmstadt.

purpose, forecasts obtained from a suitable process model, usually comprising partial differential equations (PDEs), are repeatedly combined with measurements provided by a wireless sensor network. Due to the development in the field of autonomous unmanned vehicles, mobile sensor-carrying vehicles are more and more applied in this context [4]. They are not only capable of taking measurements at different locations, also, they are able to adapt their movement online based on the current process estimates, leading to a concept called Dynamic Data-Driven Application System (DDDAS) [2]. In this way, measurements can be obtained at locations that are likely to provide a higher amount of information for process estimation.

A number of monitoring approaches have been proposed in the recent past concerning the estimation part as well as the question of how to optimally control the vehicles [17, 20, 3, 15, 5]. These approaches are all based on a centralized sensor network. A central supercomputer performs the model prediction, receives the measurements of the sensing vehicles, fuses forecast and prediction, generates suitable control inputs, and sends the input back to the vehicles. Such approaches demand for vast communication ranges, own a central point of failure, and lack scalability with respect to larger problem domains and a larger number of vehicles.

For this reason, decentralized approaches, in which the central processing node is omitted, with information being processed on-board the sensor platforms and being exchanged with local neighbors, are a better choice. While several general propositions concerning decentralized estimation and control strategies have been made [11, 12], the handling of PDE-based estimation and its relation to vehicle control remain challenging in the considered applications. Discretized PDE-models usually require the solution of high-dimensional problems, which is not at all tailored for onboard computational units with limited computational power. One possible alleviation for this challenge is provided by the application of reduced order models, where the full order dynamics are projected onto a lower dimensional space. Approaches using reduced order models have been applied in [14] and [16] and have shown good performance for relatively small problem domains and a low number of sensor-carrying vehicles. However, if the spatial dimension grows and the number of vehicles increases, further simplifications become necessary to still meet the computational restrictions.

A suitable simplification, which further supports decentralization, is the application of domain decomposition. The global problem domain is decomposed into several subdomains and each sensor vehicle is assigned to the subdomain it is located in. Every node only performs calculations concerning its own local subdomain and communicates with sensor nodes in neighboring subdomains to resolve boundary issues. In this way, an enormous amount of computational time can be saved.

If domain decomposition is applied, an adequate partitioned algorithm for simulation, data assimilation, and vehicle control has to be designed. While there is rich literature on domain decomposition regarding the simulation of dispersion processes [19], only few work has been published in the field of data assimilation on partitioned domains. Most of the work in this field focusses on sparse interconnections [10] between the models. On the other hand, problems are considered that arise from the discretization of PDEs, for which an overlapping domain decomposition method with Kalman Filter is applied [1]. However, the propagation of the error variance from one subdomain to another, an important process in the context of adaptive monitoring, is avoided.

In this work, a new scalable decentralized dynamic data-driven monitoring approach working on partitioned domains is presented. The computational effort is significantly reduced since every node only has to maintain a model of its own subdomain and only has to integrate measurements stemming from its own and from adjacent subdomains. Furthermore, the applied methods provide a reduced communication effort: Only low-dimensional vectors and matrices have to be communicated. A decentralized partitioned prediction and update method based on the Ensemble Square Root Filter is proposed that is capable of propagating neighboring measurement information as well as uncertainty over the interfaces of the subdomains. It is combined with a cooperative feedback controller that guides the sensing vehicles to informative measurement locations within the subdomain and also permits the movement of a vehicle from one subdomain to a neighboring subdomain where it might be more useful.

2 Centralized Monitoring Approach

The basic methods of the dynamic data-driven monitoring approach are introduced in this section on the basis of a centralized approach presented in previous work [15, 5, 16]. Based thereupon, the decentralized partitioned approach is developed in Section 3.

2.1 Forecast and Observation Model

The considered dispersion process is usually modeled by a PDE so that a model forecast starting from time $t = 0$ can be obtained by solving the initial boundary value problem

$$\frac{\partial x(\mathbf{r},t)}{\partial t} + \mathcal{A}(x(\mathbf{r},t)) = f \quad \text{in } \Omega \tag{1a}$$

$$x(\mathbf{r},t) = x_D(\mathbf{r},t) \quad \text{on } \partial\Omega_D \tag{1b}$$

$$\frac{\partial x(\mathbf{r},t)}{\partial n} = d_N(\mathbf{r},t) \quad \text{on } \partial\Omega_N \tag{1c}$$

$$x(\mathbf{r},0) = x^0(\mathbf{r}) \quad \text{in } \Omega. \tag{1d}$$

In this formulation, the scalar function $x(\mathbf{r},t)$ represents the dispersed entity to be estimated with spatial vector $\mathbf{r} \in \Omega$ and time $t \in \mathbb{R}_0^+$.

The initial boundary value problem (1) can be solved by a suitable discretization method, e.g. the finite element method. In this way, the solution x is approximated using a vector $\mathbf{x} \in \mathbb{R}^n$ containing the nodal values at the positions of the nodes of the underlying grid. Combined with a time discretization method, one can express the forecast of the state vector from time t^k to time t^{k+1} as

$$\mathbf{x}^{k+1} = \mathcal{M}(\mathbf{x}^k), \tag{2}$$

where $\mathcal{M}$ describes the model forecast operator. Compared to the evolution of the true state vector $\mathbf{x}^t$, a model error is made in every forecast step due to model inaccuracies and external perturbations.

The true state vector $\mathbf{x}^t$ can be accessed in a pointwise manner with n_p direct measurements of the dispersed entity

$$\mathbf{y}^k = \mathbf{H}^k(\mathbf{x}^t)^k + \boldsymbol{\epsilon}^k, \tag{3}$$

with the sensor model matrix $\mathbf{H} \in \mathbb{R}^{n_p \times n}$ depending on the location $\mathbf{r}^k$ of the measurements. Furthermore, an observation error $\boldsymbol{\epsilon}^k$ is made, which is also assumed to be Gaussian with known covariance $\mathbf{R}^k$. The model error covariance matrix $\mathbf{R}^k$ is diagonal since it is additionally assumed that observations are uncorrelated.

2.2 Data Assimilation

To combine the model forecast (denoted by the superscript $(\cdot)^f$ in the following) and the measurements of the sensor network, a suitable data assimilation method has to be used. It is recommendable to not only consider the mean estimate $\mathbf{x}$, but also its error covariance matrix

$\mathbf{P}$. The update (superscript $(\cdot)^a$) step of the commonly used Kalman Filter reads

$$\mathbf{x}^a = \mathbf{x}^f + \mathbf{K}(\mathbf{y} - \mathbf{H}\mathbf{x}^f) \tag{4}$$

$$\mathbf{P}^a = (\mathbf{I} - \mathbf{K}\mathbf{H})\mathbf{P}^f, \tag{5}$$

with the identity matrix $\mathbf{I}$ and the Kalman gain

$$\mathbf{K} = \mathbf{P}^f(\mathbf{H})^T \left(\mathbf{H}\mathbf{P}^f(\mathbf{H})^T + \mathbf{R}\right)^{-1}, \tag{6}$$

which is chosen such that the determinant of the analysis error covariance matrix $\mathbf{P}^a$ is minimized. However, the Kalman Filter cannot be applied practically here since the error covariance matrix $\mathbf{P}$ has to be forecasted along with the state estimate resulting in an expensive multiplication of high-dimensional matrices.

Instead, ensemble methods that represent the covariance $\mathbf{P}$ implicitely by a set of different state vectors $\{\mathbf{x}^{(i)}\}_{i=1}^{n_s}$ with $n_s \ll n$ are used. In this way, expensive matrix multiplications can be avoided and (2) is used to forecast every ensemble member $\mathbf{x}^{(i)}$ to observation time. With the use of the ensemble, the mean state estimate can be determined by the mean of the sample

$$\bar{\mathbf{x}}^f = \frac{\sum_{i=1}^{n_s} \mathbf{x}^{f(i)}}{n_s}, \tag{7}$$

whereas the error covariance matrix is the sample covariance

$$\mathbf{P}^f = \frac{\sum_{i=1}^{n_s} (\mathbf{x}^{f(i)} - \bar{\mathbf{x}}^f)(\mathbf{x}^{f(i)} - \bar{\mathbf{x}}^f)^T}{n_s - 1} = \mathbf{X}^f(\mathbf{X}^f)^T \tag{8}$$

with the $n \times n_s$ matrix square root $\mathbf{X}^f$ of $\mathbf{P}^f$. In the analysis, the updated ensemble mean $\bar{\mathbf{x}}^a$ can be calculated from (7) and (8) using the Kalman update (4) and (5). As (8) should also hold after the analysis, i.e. $\mathbf{P}^a = \mathbf{X}^a(\mathbf{X}^a)^T$, the analysis ensemble has to be determined adequately. Inserting (8) in (5) yields

$$\mathbf{X}^a = \mathbf{X}^f\mathbf{T} \tag{9}$$

with a transformation matrix $\mathbf{T}$ that depends on the ensemble square root method to be used. In this work, the direct Ensemble Square Root Filter [18] is used so that the matrix $\mathbf{T}$ can be obtained from

$$\mathbf{T}\mathbf{T}^T = \mathbf{I} - (\mathbf{X}^f)^T\mathbf{H}^T(\mathbf{H}\mathbf{X}^f(\mathbf{X}^f)^T\mathbf{H}^T + \mathbf{R})^{-1}\mathbf{H}\mathbf{X}^f. \tag{10}$$

2.3 Vehicle Control

To considerably improve the process estimate, the measurements should be taken at the currently most informative measurement locations. Such profitable locations can, for example, be characterized by a high error variance $\mathbf{P}^a_{jj}$, $j \in \{1, \dots, n\}$. Thus, a suitable control law should be designed for the vehicles so that measuring along the vehicles' trajectories minimizes the trace of the error covariance matrix $\mathbf{P}^a$. Finding such optimal sensor trajectories would require the solution of an optimal control problem subject to vehicle dynamics, the evolution of the covariance matrix and further constraints. For the considered applications, this approach is computationally much too expensive, so that an alternative sub-optimal control approach seems to be more appropriate.

The approach consists of two parts. First, suitable measurement locations $\mathcal{R} = \{\mathbf{r}_1, ..., \mathbf{r}_{n_p}\}$ are identified in a sequential procedure. For simplicity, the sequential procedure considers the diagonal matrix $\mathrm{diag}\{\mathbf{P}^a_{jj}\}_{j=1}^n$ instead of the full matrix $\mathbf{P}^a$. The location $\mathbf{r}_{\max}$ belonging to $\max_{j\in\{1,\dots,n\}} \mathbf{P}^a_{jj}$ is chosen as the first target point $\mathbf{r}_1$. Then, the error covariance $\mathbf{P}^a$ is updated using (9) with a sensor model matrix $\tilde{\mathbf{H}}$ representing measurements at all already chosen target points $\mathcal{R}$. In this way, clustering of target points can be avoided. The procedure is repeated until the number of target points corresponds to the number of vehicles. This algorithm does

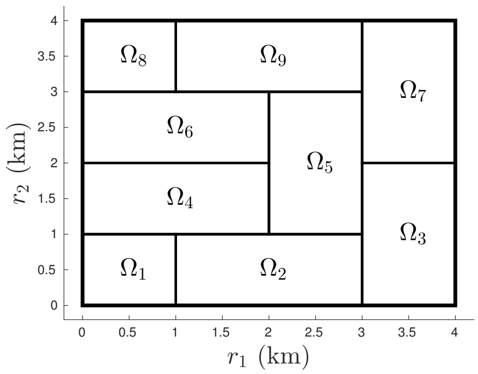


Figure 1: Exemplary domain decomposition

not require the direct computation of the matrix $\mathbf{P}^a$ since $\mathbf{P}^a_{jj} = \sum_{i=1}^{n_s} (\mathbf{X}^a_{ji})^2$.

In the second step, a cooperative feedback vehicle controller based on a mixed logical dynamical formulation is applied in a model predictive control fashion to find the control inputs such that the prescribed target points are reached [15].

3 Decentralized Monitoring Approach

To gain scalability and to avoid a central point of failure as well as vast communication requirements, the aforementioned centralized procedure is transformed into a decentralized monitoring approach. The idea is that every sensor platform computes its own model forecasts, assimilates the data obtained from own and neighboring measurements and calculates its own control inputs. However, the methods described before cannot be simply shifted to the processing unit on-board the vehicles. Onboard computational capabilities are limited and with higher-dimensional state vectors as well as with larger number of vehicles, the solution of the problem takes too much time. Hence, the idea is to apply domain decomposition and assign every vehicle to a subdomain so that only local tasks have to be fulfilled.

3.1 Domain Decomposition and Assumptions

Applying domain decomposition means sudividing the domain Ω into n_d subdomains Ω_i: $\Omega = \cup_{i \in \{1,\dots,n_d\}} \Omega_i$. In the context of this work, a non-overlapping domain decomposition is applied, i.e. $\Omega_i \cap \Omega_j = \partial\Omega_i \cap \partial\Omega_j$. An exemplary decomposition is depicted in Figure 1.

The sensors are assigned to the subdomain they are located in and perform local computations in this subdomain. Computations concerning model forecasts and vehicles located far away can be avoided. If a subdomain is occupied by several vehicles, it is assumed that every node is able to communicate with all other nodes in the subdomain and that every node maintains the same estimates. Furthermore, to provide convergence of the whole approach and smoothness at the interfaces $\Gamma_{ij} = \partial\Omega_i \cap \Omega_j$ between the subdomains, communication with all the sensors in the neighboring subdomains $\mathcal{N}_i = \{j \in \mathcal{D} : i \neq j \wedge \Gamma_{ij} \neq \emptyset\}$ is possible.

3.2 Forecast

Instead of solving the global initial boundary value problem (1), only the local model defined on the subdomain Ω_i has to be maintained by every sensor node. As the area of the subdomain Ω_i is much smaller than the area of the original domain Ω, the resulting state vector $\mathbf{x}_i$ has also a much smaller dimension and the state forecast can be obtained in a much shorter amount of time. However, one has to account for the effects stemming from neighboring domains, i.e. one has to specify the necessary conditions at the interfaces Γ_{ij}, which determine which data is exchanged between sensors on neighboring subdomains.

A popular method is the Dirichlet-Neumann method [19], in which a Dirichlet condition is used at the interface in one subdomain and a Neumann condition at the same interface in the neighboring subdomain. Thereby, the conditions mutually depend on the solution at the interfaces of the neighboring subdomains. For reasons of efficiency, this interdependence can be removed by applying the damped Adaptive Dirichlet-Neumann method [6]. In this case, the interface boundaries are assumed to be either outflow boundaries Γ_{ij}^{out} or inflow boundaries Γ_{ij}^{in}. While at the outflow boundaries a homogeneous Neumann condition is set, the Dirichlet condition at the inflow boundaries prescribes the solution of the neighboring subdomain in upstream direction at the interface. In this way, the subproblems become decoupled in each time step and the following problem has to be solved on every subdomain

$$\frac{\partial x}{\partial t} + \mathcal{A}(x) = f \quad \text{in } \Omega_i \tag{11a}$$

$$x = x_j \quad \text{on } \Gamma_{ij}^{\text{in}} \tag{11b}$$

$$\frac{\partial x}{\partial n} = 0 \quad \text{on } \Gamma_{ij}^{\text{out}}, \tag{11c}$$

where x_j is the solution of the neighboring subdomain Ω_j. Additionally, the boundary conditions (1b) and (1c) as well as the initial condition (1d) concerning the considered subdomain have to be obeyed. For advection-dominiated problems, this solution provides a sufficiently accurate approximation of the solution on the global domain.

The discretized global state vector computed with the partitioned approach is represented by the vector $\tilde{\mathbf{x}} = \text{col}\{\mathbf{x}_i\}_{i=1}^{n_d}$ and the vector $\mathbf{x}_i^{\Gamma_{ij}}$ containing all the boundary values at the interface Γ_{ij} has to be sent to the neighboring subdomain Ω_j in case of an outflow boundary or received from that subdomain in case of an inflow boundary.

3.3 Data Assimilation

In the analysis, every sensor node computes the update on its own local model incorporating all measurements of the own subdomain as well as all measurements from all neighboring subdomains. It is assumed that measurements in all other subdomains are far away and do not influence the analysis state in the local subdomain. Besides a gain in efficiency, the local data assimilation method comes along with another huge benefit. The limited ensemble size can produce spurious correlations between distant locations in $\mathbf{P}^f$ causing unphysical adjustments of model states far away from the actual observation. To suppress these correlations, local ensemble update methods that only consider observations in a vicinity of the respecting grid point have been developed [13, 8]. Hence, the decentralized data assimilation method proposed in this work is closely related to these methods as well as to the parallel ensemble filters proposed in [9, 7], but is considered for a decentralized sensor structure.

Partitioning the domain also means partitioning the ensemble set. The global partitioned error covariance square root can be defined as $\tilde{\mathbf{X}} = \text{col}\{\mathbf{X}_i\}_{i=1}^{n_d}$. Thus, each column of $\tilde{\mathbf{X}}$ represents a state vector, which is continuous at the interfaces. The observation error covariance $\tilde{\mathbf{R}} = \text{diag}\{\mathbf{R}_i\}_{i=1}^{n_d}$ and observation matrix $\tilde{\mathbf{H}} = \text{blockdiag}\{\mathbf{H}_i\}_{i=1}^{n_d}$ are partitioned accordingly.

To update the local ensemble mean, the local Kalman gain $\mathbf{K}_i$ is needed. Inserting (8) in (6) yields

$$\mathbf{K}_i = \mathbf{X}_i^f (\tilde{\mathbf{X}}^f)^T \tilde{\mathbf{H}}^T (\tilde{\mathbf{H}} \tilde{\mathbf{X}}^f (\tilde{\mathbf{X}}^f)^T \tilde{\mathbf{H}}^T + \tilde{\mathbf{R}})^{-1}. \tag{12}$$

Introducing localization, only measurements in the local vicinity of domain Ω_i are important for the update, i.e. one only needs $\tilde{\mathbf{X}}_i^{fc} = \mathrm{col}\{\mathbf{X}_i : i = j \vee i \in \mathcal{N}_i\}$, $\tilde{\mathbf{R}}_i^c = \mathrm{diag}\{\mathbf{R}_i : i = j \vee i \in \mathcal{N}_i\}$ and $\tilde{\mathbf{H}}_i^c = \mathrm{diag}\{\mathbf{H}_i : i = j \vee i \in \mathcal{N}_i\}$. The Kalman gain can then be computed as

$$\mathbf{K}_i = \mathbf{X}_i^f \mathbf{Y}_i^T (\mathbf{Y}_i \mathbf{Y}_i^T + \mathbf{R}_i^c)^{-1}, \tag{13}$$

with the $n_{pi} \times n_s$ matrix $\mathbf{Y}_i = \mathbf{H}_i^c \mathbf{X}_i^{fc} = \mathrm{col}\{\mathbf{H}_i \mathbf{X}_i^{fc} : i = j \vee i \in \mathcal{N}_i\}$. This means that, to compute the Kalman gain, only relatively small-sized entities have to be exchanged between sensors on neighboring domains: The observation error covariance matrix $\mathbf{R}_i$ and the forecast covariance at measurement locations $\mathbf{Y}_i$. Furthermore, to perform the update step (4), the innovation $\tilde{\mathbf{d}}^c = \mathrm{col}\{\mathbf{y}_i - \mathbf{H}_i \mathbf{x}_i : i = j \vee i \in \mathcal{N}\}$ is needed.

For the update of the ensemble, no further exchange is needed. The update equations (9) and (10) only require $\mathbf{X}_i^c \mathbf{H}_i^c$, i.e. $\mathbf{Y}_i$, and $\mathbf{R}_i^c$ to perform the local update.

3.4 Vehicle Control

To compute the control input vector, the vehicle controller described in the previous section is applied on-board every vehicle. In contrast to the centralized case, though, only the sensor nodes that are located in the same subdomain are considered. Hence, the number of vehicles and target points that have to be involved in the calculation is drastically reduced. Having solved the optimization problem, every vehicle only applies the control input that has been calculated for itself. The consideration of the other vehicles in the domain, however, leads to cooperation in the task of getting to the target points.

The considered applications involve a dynamic process and so it might be ineffective to work with a fixed assignment of sensors to subdomains. From a sensor's perspective, a measurement in an adjacent subdomain could be much more suitable than in the own subdomain, e.g. when the uncertainty in the neighboring subdomain is extremely high, while the estimates in the own subdomain are rather good. Therefore, an adaptive method in which, under specific conditions, the sensors are allowed to move from one subdomain to another should be implemented.

Introducing another type of target point can solve this problem. From time to time, the total subdomain variance $V_i = \mathrm{trace}(\mathbf{P}_i)$ is exchanged between neighbors. If the following condition holds for subdomains Ω_i and Ω_j

$$\frac{V_i}{n_{pi} - 1} < \frac{V_j}{n_{pj} + 1}, \tag{14}$$

a sensor of subdomain Ω_i should move to subdomain Ω_j. In that case, the last target point in $\mathcal{R}$ is replaced by a target point on the boundary Γ_{ij}. As soon as this target point is reached by a vehicle from subdomain Ω_i, it moves to subdomain Ω_j and is supplied with the necessary domain information (state ensemble, forecast model, etc.) by the sensor nodes of domain Ω_j.

4 Results

To evaluate the proposed partitioned monitoring approach, a process governed by the linear homogeneous advection-diffusion equation

$$\frac{\partial x}{\partial t} + \frac{\partial (v_1 x)}{\partial r_1} + \frac{\partial (v_2 x)}{\partial r_2} - D_1 \frac{\partial^2 x}{\partial r_1^2} - D_2 \frac{\partial^2 x}{\partial r_2^2} = 0 \tag{15}$$

is considered on the two-dimensional domain $\Omega = [0\,\mathrm{km}, 4\,\mathrm{km}] \times [0\,\mathrm{km}, 4\,\mathrm{km}]$. The scenario can be interpreted as the two-dimensional aerial dispersion of a chemical pollutant with concentration x. Typical velocity parameters for gentle wind are chosen ($v_1 = 3\,\mathrm{ms}^{-1}$, $v_2 = 2\,\mathrm{ms}^{-1}$, $D_1 = D_2 = 20\,\mathrm{m}^2\mathrm{s}^{-1}$) so that the process is advection-dominated as demanded. Besides the model error, which is accounted for by slight perturbations of the velocity, the true initial concentration is unknown. Thus, the estimation process starts with a rather bad initial estimate, chosen in this scenario to be uniformly zero, $x^0(\mathbf{r}) \equiv 0$. The corresponding error covariance is formed by a Gaussian kernel function, whose center is located at $(1.6\,\mathrm{km}, 1.5\,\mathrm{km})^T$.

The global domain comprises approximately 10,000 nodes and is decomposed into nine subdomains according to Figure 1. To generate measurements and to assess the performance of the proposed monitoring approach, the true solution starting from the true initial conditions is also simulated (twin experiment). Two sensors are initially located in every subdomain.

At first, the partitioned prediction and update procedure is briefly evaluated by omitting the vehicle control and considering a static sensor network instead. While the simplified forecast with the assumption of a homogeneous Neumann boundary condition at the interfaces between subdomains inevitably leads to an additional model error, even for advection-dominated flows, no additional error compared to its centralized counterpart is introduced by the update step if there are no correlations between non-adjacent subdomains. Figure 2 shows the root mean square error between true state and estimate over time – once computed on the full domain and once using the new partitioned approach. The deviation in error between both approaches is small and does, due to repeated measurement updates, hardly grow over time. Thus, the prediction and update scheme can be applied without compunction in the considered scenario.

Now, the sensors are mounted onto vehicles, modeled with double-integrator dynamics, enabling the dynamic data-driven monitoring approach proposed in this work.

The resulting dynamic data-driven estimate at $t = 100\,\mathrm{s}$ is depicted in Figure 4(a). Compared to the true solution in Figure 4(b), deviations can be only noticed at the sides of the concentration peaks and in the magnitude of them. In total, the monitoring approach succeeds in providing a rather good qualitative estimate after a rather short amount of time.

The good performance is also reflected in the evolution of the root mean square error over time plotted in Figure 2. With the first measurements, the estimate is already strongly improved. The next phase mainly consists of the redistribution of the vehicles so that the estimate temporarily gets worse, but the adaptation leads to a further decrease in error in the following. To compare these results, the estimation error of the original centralized approach described in Section 2 is also evaluated. The vehicles are not restricted to stay on a local domain and follow a global control law guiding them to more informative locations. Thus, the error obtained with this approach is smaller than with the partitioned approach, especially when the decentralized approach is mainly occupied with redistributing the vehicles. However, the results obtained with the partitioned approach are not much worse, as soon as the sensors are distributed over the subdomains in a reasonable way. Taking into account that the centralized approach requires much more computation time due to the larger problem dimension and the larger multi-vehicle system to be considered, further revaluates the results obtained with the partitioned approach. For the considered application and decomposition, the speedup of the proposed approach regarding state forecast amounts to approximately 10, whereas for the speedup regarding vehicle control, an even higher gain can be achieved. Moreover, compared to the solution obtained with the static sensor networks discussed before, the dynamic data-driven solution is significantly better.

The adaptive redistribution ability of the vehicles is shown by Figure 3, depicting a histogram of the number of sensors on every domain at $t = 100\,\mathrm{s}$. As the main concentration field is located in the lower and middle part of the domain, a lot of sensors in the top part are hardly needed but are strongly required in the lower part. Thus, adapting to the current uncertainty, sensors move from the top part to the subdomains located in the bottom and middle part of the domain. At $t = 100\,\mathrm{s}$, the subdomains at the top and the small subdomain Ω_1 only contain one sensor unit, while the subdomains in the lower and middle part contain two or more.

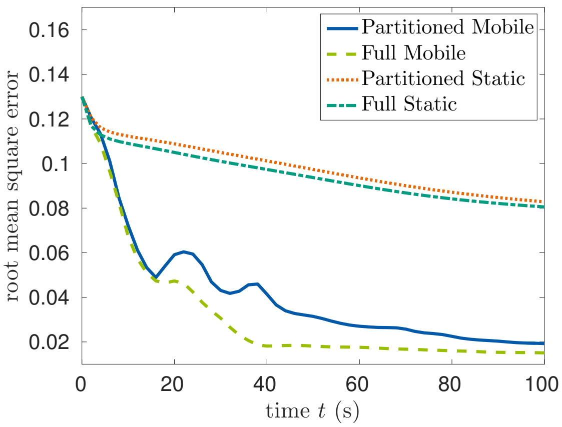


Figure 2: Root mean square error over time for partitioned and full monitoring with static and mobile sensor platforms

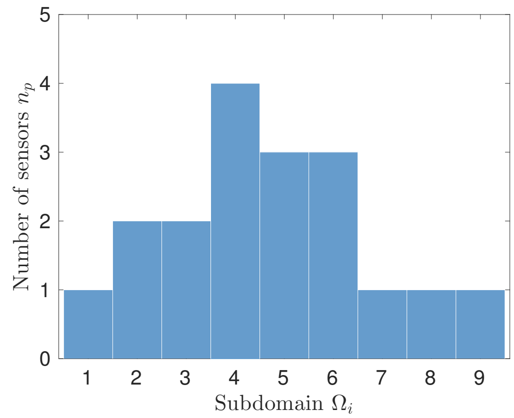
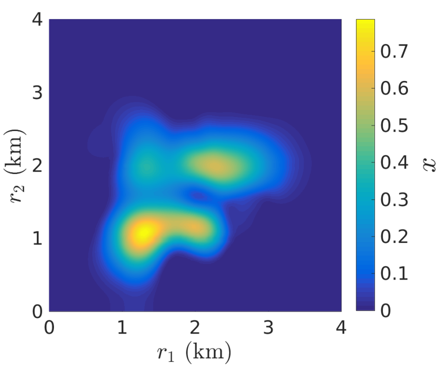
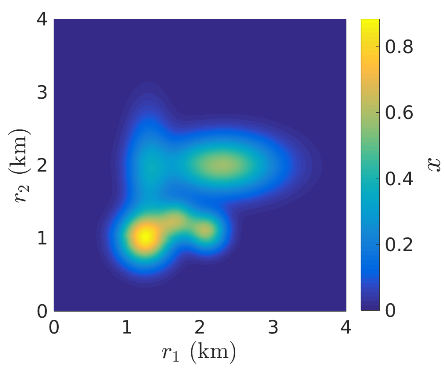


Figure 3: Histogram showing the number of sensors per subdomain at $t = 100\,\mathrm{s}$



(a) Estimated concentration



(b) True concentration

Figure 4: Concentration field at $t = 100\,\mathrm{s}$

5 Conclusion

A new decentralized dynamic data-driven strategy for online estimating dispersion processes has been proposed. To attain a strategy that is computationally tractable, even when performed on-board the sensor platforms, domain decomposition is applied effecting every sensor node only to perform calculations concerning the local process and sensor model. A decentralized prediction and update method based on damped Adaptive Dirichlet-Neumann and a partitioned variant of the Ensemble Square Root Filter requiring only minimum communication with sensors on neighboring domains ensures global convergence. The applied vehicle controller not only guides the vehicles cooperatively to interesting measurement locations within the subdomains but also allows a dynamic redistribution concerning exchanges of vehicles between subdomains. Compared to its centralized counterpart, the estimation results obtained with the new approach rank only a little behind, whereas the computation time can be reduced significantly. For a further reduction of computation time in the future, the additional application of reduced order models could be helpful in context of larger problem dimensions and a more heuristic controller would help in context of a larger number of vehicles.

References

[1] G. Battistelli, L. Chisci, N. Forti, G. Pelosi, and S. Selleri. Distributed finite element kalman filter. In *Proceedings of the 2015 European Control Conference (ECC)*, pages 3695–3700. IEEE, 2015.

[2] F. Darema. Dynamic data driven applications systems: A new paradigm for application simulations and measurements. In *International Conference on Computational Science (ICCS)*, pages 662–669. Springer, 2004.

[3] M. Demetriou and D. Ucinski. State estimation of spatially distributed processes using mobile sensing agents. In *Proceedings of the American Control Conference*, pages 1770–1776, 2011.

[4] M. Dunbabin and L. Marques. Robots for environmental monitoring: Significant advancements and applications. *IEEE Robotics Automation Magazine*, 19(1):24–39, 2012.

[5] J. Euler, T. Ritter, S. Ulbrich, and O. von Stryk. Centralized ensemble-based trajectory planning of cooperating sensors for estimating atmospheric dispersion processes. In *Dynamic Data-Driven Environmental Systems Science*, number 8964 in Lecture Notes in Computer Science, pages 322–333. Springer International Publishing, 2015.

[6] F. Gastaldi, L. Gastaldi, and A. Quarteroni. Adn and arn domain decomposition methods for advection-diffusion equations. In *Proceedings of the 9th International Conference on Domain Decompositon Methods in Science and Engineering. Wiley & Sons*, 1997.

[7] P. L. Houtekamer and H. L. Mitchell. A sequential ensemble Kalman filter for atmospheric data assimilation. *Monthly Weather Review*, 129(1):123–137, 2001.

[8] B. R. Hunt, E. J. Kostelich, and I. Szunyogh. Efficient data assimilation for spatiotemporal chaos: A local ensemble transform Kalman filter. *Physica D: Nonlinear Phenomena*, 230(1-2):112–126, 2007.

[9] C. L. Keppenne. Data assimilation into a primitive-equation model with a parallel ensemble kalman filter. *Monthly Weather Review*, 128(6):1971–1981, 2000.

[10] U. A. Khan and J. M. Moura. Distributing the kalman filter for large-scale systems. *Signal Processing, IEEE Transactions on*, 56(10):4919–4935, 2008.

[11] J. Manyika and H. Durrant-Whyte. *Data Fusion and Sensor Management: a decentralized information-theoretic approach.* New York: Ellis Horwood, 1995.

[12] A. Mutambara. *Decentralized estimation and control for multisensor systems.* CRC press, 1998.

[13] E. Ott, B. R. Hunt, I. Szunyogh, A. V. Zimin, E. J. Kostelich, M. Corazza, E. Kalnay, D. Patil, and J. A. Yorke. A local ensemble kalman filter for atmospheric data assimilation. *Tellus A*, 56(5):415–428, 2004.

[14] L. Peng, M. Silic, and K. Mohseni. A DDDAS plume monitoring system with reduced Kalman Filter. In *International Conference on Computational Science (ICCS)*, volume 51, pages 2533–2542. Elsevier, 2015.

[15] T. Ritter, J. Euler, S. Ulbrich, and O. von Stryk. Adaptive observation strategy for dispersion process estimation using cooperating mobile sensors. In *Proceedings of the 19th IFAC World Congress*, pages 5302 – 5308, 2014.

[16] T. Ritter, J. Euler, S. Ulbrich, and O. von Stryk. Decentralized dynamic data-driven monitoring of atmospheric dispersion processes. *Procedia Computer Science*, 80:919–930, 2016.

[17] Z. Song, Y. Chen, J. Liang, and D. Ucinski. Optimal mobile sensor motion planning under nonholonomic constraints for parameter estimation of distributed systems. *International Journal of Intelligent Systems Technologies and Applications*, 3(3/4):277–295, 2007.

[18] M. K. Tippett, J. L. Anderson, C. H. Bishop, T. M. Hamill, and J. S. Whitaker. Ensemble square root filters. *Monthly Weather Review*, 131(7):1485–1490, 2003.

[19] A. Toselli and O. Widlund. *Domain decomposition methods: algorithms and theory*, volume 34. Springer, 2005.

[20] D. Zhang, C. Colburn, and T. Bewley. Estimation and adaptive observation of environmental plumes. In *Proceedings of the American Control Conference*, pages 4281–4286, 2011.