Cooperative N-Boundary Tracking in Large Scale Environments

Juliane Euler, Andreas Horn, Dominik Haumann, Jürgen Adamy, and Oskar von Stryk

Abstract—**Monitoring in large scale environments is a typical mission in cooperative robotics. This task requires the exploration of a huge domain by a generally small number of sensor equipped mobile robots. As time restrictions prohibit an exhaustive global search, a sampling strategy is required that allows an efficient spatial mapping of the environment.**

This paper proposes an adaptive sampling strategy for efficient simultaneous tracking of multiple concentration levels of an atmospheric plume by a team of cooperating unmanned aerial vehicles (UAVs). The approach combines uncertainty and correlation-based concentration estimates to generate sampling points based on already gathered data. The adaptive generation of sampling locations is coupled to a distributed model-predictive controller for planning optimal vehicle trajectories under collision and communication constraints. Simulation results demonstrate that connectivity of all involved vehicles can be maintained and an accurate reconstruction of the plume is obtained efficiently.

I. INTRODUCTION

The use of robotic systems in monitoring scenarios has been increasingly considered in recent years. The objective is to gather information on environmental phenomena by a group of sensor equipped robots. Applications include tracking of oil fields under water, monitoring of disaster areas on the ground, or aerial monitoring of airborne contaminants.

This work focuses on cooperative monitoring in large scale environments, which poses additional challenges compared to small or medium scale environments (cf. [1]). In the latter cases, a single robot may fulfill the monitoring task sufficiently. However, monitoring in large scale environments requires that a rather huge area needs to be covered by a generally small number of robots. As a consequence, an exhaustive global search cannot be performed. Instead, a suitable sampling strategy is required that *adaptively* determines new sampling locations for the robots, which (i) promise a maximum information gain in minimum time, (ii) are sufficiently dense to permit an accurate reconstruction of the considered phenomenon, and (iii) account for the robots' physical capabilities and constraints. Requirement (iii) implies a direct coupling of the adaptive sampling

J. Euler and O. von Stryk are with the Simulation, Systems Optimization & Robotics Group, Technische Universität Darmstadt, Hochschulstr. 10, 64289 Darmstadt, Germany, {euler, stryk}@sim.tu-darmstadt.de.

A. Horn, D. Haumann and J. Adamy are with the Institute of Automatic Control & Mechatronics, Technische Universität Darmstadt, Landgraf-Georg-Str. 4, 64283 Darmstadt, Germany, ahorn@iat.tu-darmstadt.de, {dhaumann, jadamy}@rtr.tu-darmstadt.de.

approach to a cooperative controller that generates efficient motion trajectories for the multi-robot system.

A typical large scale mission in environmental monitoring is the detection of atmospheric plumes. In particular, the surveillance of a plume's perimeter or the tracking of a certain concentration level may require exploration of a vast area. Here, the use of multiple robots for cooperative data-gathering offers an obvious benefit and has been considered by a number of scientists.

[2] and [3] use multiple autonomous vehicles to track a boundary based on Bayesian probability theory. However, cooperative aspects and the vehicle dynamics are not taken into account when choosing new sampling points. In [4], motion dynamics are considered, but the vehicles are coordinated in terms of collision avoidance only. In [5], multiple vehicles move in a formation along the boundary. In addition to the boundary, one gains information on the gradient of the tracked concentration level. Predefined motion patterns for a single vehicle are used in [6] and [7] for adaptive plume mapping and boundary tracking.

In [8], multiple vehicles steadily move on a polygonal approximation of a boundary while maintaining distances of equal length in a fixed ring topology. In [9], the team's workload is balanced by distributively assigning boundary parts to each robot. [10] proposes an entropy-based sampling method and motion coordination based on a Voronoi partition. Approaches [8]–[10] assume reliable team communication independent of the distances between the robots, but this assumption does not necessarily hold in large scale environments.

Several approaches employ the advection-diffusion equation ([11], [12]) to model the dispersion of the plume. For estimating the model parameters, the authors of [13] present a locally optimal path planning approach, but restrict their work to the single-robot case. [14] and [15] use sensors at fixed positions to gather concentration measurements. Another model-based approach is given in [16] to sample an environmental process under water.

Contribution of this note. We present an adaptive sampling strategy for efficient simultaneous tracking of multiple concentration levels of an atmospheric plume by a team of cooperating unmanned aerial vehicles (UAVs) (see Fig. 1). The adaptive generation of sampling points combines multiple information sources. It is based on the uncertainty associated with the concentration at the respective location as well as concentration estimates determined from already gathered measurement data. The strategy can flexibly be applied to various types of (not necessarily atmospheric) plumes without prior model knowledge. For simplification, time-invariant
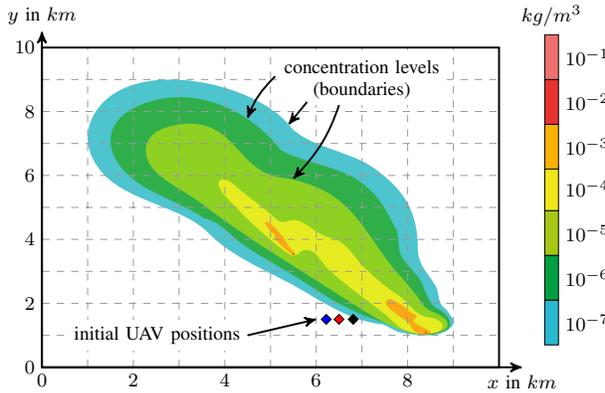
Fig. 1. Concentration levels of the dispersion of an airborne contaminant. The plume is generated by SciPuff [19].

plumes are considered in this paper. Tracking multiple concentration levels by one vehicle each efficiently provides not only an accurate estimate of the plume's perimeter, but additional information on the concentration gradient. We refer to this approach as *cooperative N-boundary tracking*.

The vehicles' task is to cooperatively visit the adaptively generated, discrete measurement locations at a fixed altitude above ground. Since the efficiency of the proposed sampling strategy depends on constant information exchange among the UAVs, stable communication has to be ensured. Therefore, in order to maintain connectivity and to avoid collisions, certain lower and upper distance limits may not be exceeded by the vehicles. Consequently, the planning of optimal vehicle trajectories results in solving a cooperative mobility problem [17]. For this purpose, the adaptive generation of sampling locations is coupled to a distributed model-predictive controller (MPC) for path planning under collision and communication constraints. The employed MPC approach is based on a *Mixed Logical Dynamical* (MLD) formulation of the considered cooperative control problem and was first proposed in [18] for a benchmark scenario from cooperative target observation.

This paper is organized as follows: In Section II, we introduce the MPC framework for optimal vehicle control under motion constraints and describe the underlying system model. In Section III, the sampling strategy generating new sampling points for each UAV is proposed. Section IV provides simulation results of the presented tracking approach for the atmospheric plume depicted in Fig. 1. A conclusion is given in Section V.

II. DISTRIBUTED PREDICTIVE CONTROL FOR A MIXED LOGICAL DYNAMICAL MODEL

A. MLD-Based Model-Predictive Control

The basic idea of the cooperative control approach employed in this paper is to set up a discrete-time linear MLD model of the considered multi-vehicle system, combine it with a suitable objective function, and solve the resulting optimal control problem

$$\min_{U_N} \ |\mathbf{P}\mathbf{x}^N| + \sum_{k=0}^{N-1} |\mathbf{Q}_1\mathbf{u}^k| + |\mathbf{Q}_2\boldsymbol{\delta}^k| + |\mathbf{Q}_3\mathbf{z}^k| + |\mathbf{Q}_4\mathbf{x}^k| \quad \text{(1a)}$$

$$\text{s.t.} \quad \mathbf{x}^{k+1} = \mathbf{A}\mathbf{x}^k + \mathbf{B}_1\mathbf{u}^k + \mathbf{B}_2\boldsymbol{\delta}^k + \mathbf{B}_3\mathbf{z}^k \quad \text{(1b)}$$

$$\mathbf{y}^k = \mathbf{C}\mathbf{x}^k + \mathbf{D}_1\mathbf{u}^k + \mathbf{D}_2\boldsymbol{\delta}^k + \mathbf{D}_3\mathbf{z}^k \quad \text{(1c)}$$

$$\mathbf{E}_2\boldsymbol{\delta}^k + \mathbf{E}_3\mathbf{z}^k \leq \mathbf{E}_1\mathbf{u}^k + \mathbf{E}_4\mathbf{x}^k + \mathbf{E}_5 , \quad \text{(1d)}$$

in an receding horizon fashion to compute optimal control inputs for each vehicle. In this problem formulation, $\mathbf{x} = [\mathbf{x}_c \ \mathbf{x}_b]^T, \mathbf{x}_c \in \mathbb{R}^{n_c}, \mathbf{x}_b \in \{0,1\}^{n_b}$, is the system state, $\mathbf{y} = [\mathbf{y}_c \ \mathbf{y}_b]^T, \mathbf{y}_c \in \mathbb{R}^{p_c}, \mathbf{y}_b \in \{0,1\}^{p_b}$, is the output vector, $\mathbf{u} = [\mathbf{u}_c \ \mathbf{u}_b]^T, \mathbf{u}_c \in \mathbb{R}^{m_c}, \mathbf{u}_b \in \{0,1\}^{m_b}$, is the control input, and $\boldsymbol{\delta} \in \{0,1\}^{r_b}$ and $\mathbf{z} \in \mathbb{R}^{r_c}$ represent auxiliary binary and continuous vectors, respectively. The prediction time step $k = 0, \ldots, N-1$ relates to the global equidistant time steps $t \in \mathbb{Z}$ according to $\mathbf{x}^k = \mathbf{x}(t+k)$. As solution of problem (1), the sequence $U_N := \{\mathbf{u}^k\}_{k=0}^{N-1}$ of control inputs is obtained.

In virtue of a model-predictive control scheme, the first element of U_N is applied to the real system, then its new state is measured for computing an updated control input sequence at the next time step t. In this manner, the prediction horizon N is shifted over time. The main advantage of this strategy is its ability to compensate modeling inaccuracies and disturbances.

The MLD framework (1b)–(1d) was proposed in [20] for modeling and controlling constrained linear systems containing interacting physical laws and logical rules. It is a powerful tool for a wide range of tasks involving cooperative mobility, especially when combined with a predictive feedback control strategy. The objective function (1a) can reflect the prioritization of different problem aspects.

Problem (1) is a mixed-integer linear *Constrained Finite Time Optimal Control* (CFTOC) problem. It can easily be transformed into a *Mixed Integer Linear Program* (MILP) at each time step of the MPC procedure. Therefore a numerically robust, efficient computation of control inputs can be performed. It guarantees global optimality within the scope of the underlying system model without strongly depending on initial guesses or bounds, as it would be the case in mixed-integer non-linear programming. The outlined cooperative control approach is detailed in [18].

B. Distributed Control for Cooperative Boundary Tracking

From a single vehicle's point of view, only an excerpt of the overall tracking problem is relevant. This subproblem involves the vehicle itself and teammates within its communication range. Moreover, the vehicles share the coordinates of their individual target points. Based on this locally available information, a vehicle's individual model-predictive controller as introduced in Section II-A then provides its optimal next move towards the current target point. For this purpose, a model of the local subsystem that includes the currently involved $\tilde{n}_V$ vehicles and their target points is set up and serves as a basis for predicting the evolution of the system state. Hence, each controller computes optimal

behavior for all involved vehicles, from which only the control input for the currently considered vehicle is actually applied.

In the current implementation of this approach, a priori the application, several MLD models are set up offline, one for each possible number of vehicles in a local environment. During the online application, an appropriate model and the corresponding controller is selected by a vehicle based on the number of teammates within communication range. In order to obtain an efficient online control strategy, a maximum number of vehicles $\tilde{n}_{V\max}$ in a model may not be exceeded. During application, if there is information on more vehicles available than the defined maximum, only those closest to the controlled vehicle are included in the subsystem model. The choice of $\tilde{n}_{V\max}$ obviously depends on the desired communication topology and influences the quality of the obtained control inputs with respect to the overall vehicle cooperation.

This distributed control approach is beneficial especially when considering a large number of vehicles and a communication topology that does not require single-hop connectivity among all of them. It is flexible in terms of the overall number of vehicles and does not depend on a central component and a permanent, stable communication with it. The vehicles' individual controllers are coupled by the given cooperation rules and the common objective. That way, trajectories can be computed efficiently and still result in a near optimal cooperative strategy.

C. MLD Model of a Subsystem

Motion: Employing the MLD framework (1b)–(1d) for modeling multi-vehicle systems permits to use all kinds of discrete-time linear motion dynamics models that can be stated in the form

$$\mathbf{x}_v^{k+1} = \mathbf{A}_v \mathbf{x}_v^k + \mathbf{B}_v \mathbf{u}_v^k , \qquad (2)$$

where $\mathbf{x}_v^k$ and $\mathbf{u}_v^k$ denote the state and control input, respectively, of vehicle $v \in \{1, \ldots, \tilde{n}_V\}$ at time step $k = 0, \ldots, N-1$.

Distances: A linear approximation d_{vs}^k of the exact Euclidean distance between a vehicle v and a sampling point (x_s, y_s) is obtained by introducing a set of inequalities

$$(x_v^k - x_s)\sin\frac{2\pi\gamma}{n_\gamma} + (y_v^k - y_s)\cos\frac{2\pi\gamma}{n_\gamma} \;\le\; d_{vs}^k, \qquad (3)$$

where (x_v^k, y_v^k) denotes the vehicle's position at time k and $\gamma = 1, \ldots, n_\gamma$. If d_{vs}^k takes the minimum value such that all inequalities (3) hold, then $d_{vs}^k \approx \sqrt{(x_v^k - x_s)^2 + (y_v^k - y_s)^2}$. Here, the accuracy of the approximation can be scaled by the constant parameter $n_\gamma \in \mathbb{N}$. The overall optimization scheme ensures that d_{vs}^k is driven to its smallest possible value.
In the following sections, we shorten expression (3) by introducing the notation $\sin_\gamma := \sin\frac{2\pi\gamma}{n_\gamma}$ and $\cos_\gamma := \cos\frac{2\pi\gamma}{n_\gamma}$.

Communication: Distances between vehicles v_i and v_j are approximated in the same manner as (3). Since the UAVs are required to stay within reach of communication, their distance to each other is limited to a maximum value of d_{com},

and a binary variable $b_{\mathrm{com},ij}^{k\gamma}$ indicates whether this condition holds at time step k:

$$b_{\mathrm{com},ij}^{k\gamma} = 0 \quad \Leftrightarrow \quad g_{1,ij}^k(\gamma) \;\le\; 0 \qquad (4)$$

for $g_{1,ij}^k(\gamma) = (x_{v_i}^k - x_{v_j}^k)\sin_\gamma + (y_{v_i}^k - y_{v_j}^k)\cos_\gamma - d_{com}$ and $v_i, v_j \in \{1, \ldots, \tilde{n}_V\}, v_i \neq v_j, \gamma = 1, \ldots, n_\gamma$. In order to develop a MLD representation of the considered system, logical statements like (4) have to be transformed into linear inequalities. Using the Big-M method from [21], (4) can be rewritten as the following sequence of linear inequalities:

$$g_{1,ij}^k(\gamma) \le M_1 b_{\mathrm{com},ij}^{k\gamma} \quad \text{and} \qquad (5)$$
$$g_{1,ij}^k(\gamma) \ge \varepsilon + (m_1 - \varepsilon)(1 - b_{\mathrm{com},ij}^{k\gamma}) , \qquad (6)$$

where $M_1 = \max(g_{1,ij}^k(\gamma))$ and $m_1 = \min(g_{1,ij}^k(\gamma))$. If all variables $b_{\mathrm{com},ij}^{k\gamma} = 0$, the above constraints represent a fully connected communication topology among the vehicles.

Collision: Introducing a constraint similar to (4) for a minimum distance d_{col} and additional sets of binary variables $b_{\mathrm{col},ij}^{k\gamma}$ and $b_{\mathrm{col},ij}^k$ assures that the vehicles do not collide:

$$b_{\mathrm{col},ij}^{k\gamma} = 1 \quad \Leftrightarrow \quad g_{2,ij}^k(\gamma) \;\le\; 0 \quad \text{and} \qquad (7)$$
$$b_{\mathrm{col},ij}^k = 1 \quad \Leftrightarrow \quad n_\gamma - \sum_{\gamma=1}^{n_\gamma} b_{\mathrm{col},ij}^{k\gamma} \;\le\; 0, \qquad (8)$$

where $g_{2,ij}^k(\gamma) = (x_{v_i}^k - x_{v_j}^k)\sin_\gamma + (y_{v_i}^k - y_{v_j}^k)\cos_\gamma - d_{col}$. The variables $b_{\mathrm{col},ij}^k$ could as well be omitted, but are used here in order to be able to penalize the violation of inequality (8) via the objective function on top of the system model. That way, (8) represents a soft constraint. The transformation of (7) and (8) into linear inequalities yields

$$g_{2,ij}^k(\gamma) \;\le\; M_2(1 - b_{\mathrm{col},ij}^{k\gamma}) , \qquad (9)$$
$$g_{2,ij}^k(\gamma) \;\ge\; \varepsilon + (m_2 - \varepsilon)b_{\mathrm{col},ij}^{k\gamma} , \qquad (10)$$
$$n_\gamma - \sum_{\gamma=1}^{n_\gamma} b_{\mathrm{col},ij}^{k\gamma} \;\le\; M_3(1 - b_{\mathrm{col},ij}^k) , \quad \text{and} \qquad (11)$$
$$n_\gamma - \sum_{\gamma=1}^{n_\gamma} b_{\mathrm{col},ij}^{k\gamma} \;\ge\; \varepsilon + (m_3 - \varepsilon)b_{\mathrm{col},ij}^k , \qquad (12)$$

where $M_2 = \max(g_{2,ij}^k(\gamma))$, $m_2 = \min(g_{2,ij}^k(\gamma))$, $M_3 = \max(n_\gamma - \sum_{\gamma=1}^{n_\gamma} b_{\mathrm{col},ij}^{k\gamma})$, and $m_3 = \min(n_\gamma - \sum_{\gamma=1}^{n_\gamma} b_{\mathrm{col},ij}^{k\gamma})$.

Objectives: The controller's essential purpose is to lead each vehicle to its assigned target location (x_s, y_s), which is represented in the objective function as minimization of the distances d_{vs}^k. At the same time, the distance limits induced by the communication and collision constraints are to be met at all times. This is ensured by penalizing the binary variables $b_{\mathrm{com},ij}^{k\gamma}$ and $b_{\mathrm{col},ij}^k$ whenever they take the value 1. In addition, the vehicles are to move at a minimum control effort, which in reality could correspond to energy consumption or other limiting factors. In summary, the cost function takes the

following form:

$$\min_{U_N} \sum_{k=0}^{N-1} \Big(q_z \sum_{v,s=1}^{\tilde{n}_V} d_{vs}^k +$$

$$q_\delta \sum_{i=1}^{\tilde{n}_V-1} \sum_{j=i}^{\tilde{n}_V} \big(b_{\text{col},ij}^k + \sum_{\gamma=1}^{n_\gamma} b_{\text{com},ij}^{k\gamma} \big) + q_u |\mathbf{u}^k| \Big) \,, \quad (13)$$

where $\mathbf{u}^k$ concatenates the vehicle control inputs $\mathbf{u}_v^k$ and q_z, q_δ, $q_u \in \mathbb{R}$ weight the different objectives according to their priorities and the best expected task performance.

Problem Size: In order to apply the MPC approach as outlined in Section II-A, the mixed-integer problem consisting of (2)–(3), (5)–(6), (9)–(12), and (13) is reformulated in form of problem (1). In this representation, the vector $\mathbf{x}^k$ contains the state of all $\tilde{n}_V$ vehicles in the subsystem as well as the coordinates of their target locations. All binary variables are contained in $\boldsymbol{\delta}^k \in \{0,1\}^{\binom{\tilde{n}_V}{2}(2n_\gamma+1)}$. $\mathbf{z}^k \in \mathbb{R}^{\tilde{n}_V}$ comprises the approximated distances d_{vs}^k. The vector $\mathbf{u}^k$ summarizes the vehicle control inputs. The overall problem comprises $\tilde{n}_V \cdot n_\gamma + 4 \cdot n_\gamma \cdot \binom{\tilde{n}_V}{2} + 2 \cdot \binom{\tilde{n}_V}{2}$ linear inequality constraints.

III. SAMPLING STRATEGY

The adaptive sampling strategy presented in this section generates sampling points such that the requirements (i)–(iii) stated in Section I are met. First, the underlying representation of the work area and the process of performing a measurement at a sampling point are described. Afterwards, the adaptive sampling algorithm is proposed.

A. Map Representation

Each vehicle has its own representation of the bounded work area $\mathcal{G} \subset \mathbb{R}^2$. The work area is represented as a grid map consisting of discrete cells. A cell at position (x,y) stores a Gaussian distribution defined by the expected concentration value $c(x,y)$ and the variance $\text{var}(x,y)$. The distribution describes the knowledge about the dispersion process in the entire cell. The variance is regarded in terms of the uncertainty of the contaminant concentration of the respective cell. Since no knowledge about the contaminant distribution is available a priori, the distributions in all cells are initially set to a mean of $c_{\text{init}}(x,y) = 0 \frac{\text{kg}}{\text{m}^3}$ and a variance of $\text{var}_{\text{init}}(x,y) = \infty$.

Vehicles within communication range instantly share their sampling data, such that synchronized data is available for the whole team. Communication is assumed to be free of delays, bandwidth restrictions, or other limiting factors.

B. Sample Processing

A vehicle is required to stay at a sampling location $(x_s, y_s) \in \mathcal{G}$ for N_m successive time steps to successfully process a sample. In our model, concentration samples $c_s = c(x_s, y_s)$, taken at a sampling point (x_s, y_s), are distorted by Gaussian measurement noise $\mathcal{N}(\mu, \sigma^2)$ with mean μ and variance σ^2. In order to reduce the effects of the noise, we apply a discrete Kalman filter.

Since spatial concentration dispersions of airborne contaminants typically change smoothly, a certain correlation of adjacent measurements is assumed. To this end, we introduce a fixed correlation coefficient $r_\mathcal{I} = \text{cov}(c_s, c^*)$ between two grid cells with concentration samples c_s and c^*. We assume, that this correlation holds for a certain area $\mathcal{I} \subset \mathcal{G}$ around a sampling point (x_s, y_s), termed as *impact area*. In our case, we assume a rectangular impact area which is determined by the *impact range* d_{imp}:

$$\mathcal{I} := [x_s - d_{\text{imp}}, x_s + d_{\text{imp}}] \times [y_s - d_{\text{imp}}, y_s + d_{\text{imp}}]. \quad (14)$$

This way, we can infer information about the contaminant concentration and the respective variances of surrounding locations from a single sampling point (x_s, y_s), which expedites the overall sampling process.

C. Adaptive Sampling Algorithm

Generation of new sampling points is based on shared information that already has been acquired throughout the sampling process. Since the amount of information grows over time, the sampling process is called *adaptive*. The algorithm is implemented on each UAV and individually generates sampling points that track the assigned concentration level l_c.

New sampling locations (x_s, y_s) within $\mathcal{G}$ are selected depending on multiple information types. The strategy accounts for the knowledge on the concentration distribution as well as for the respective variances, which results in a two-stage approach.

Step 1 (Maximum Variance): In order to determine a sampling point providing the highest information gain, a set $\mathcal{C}$ of n_c candidate sampling points $(x_{c,i}, y_{c,i})$, $i = 1, \ldots, n_c$, is selected by identifying the n_c maximum variance values (cf. Fig. 2). This operation is constrained to the *action area* $\mathcal{R}_{\text{act}} \subset \mathcal{G}$ around the position (x_v, y_v) of a vehicle specified by the *action range* d_{act}:

$$\mathcal{R}_{\text{act}} := [x_v - d_{\text{act}}, x_v + d_{\text{act}}] \times [y_v - d_{\text{act}}, y_v + d_{\text{act}}]. \quad (15)$$

As the MPC procedure predicts the vehicles' motion for N time steps ahead, it has to be assured, that a new sampling point is reachable within that time. Due to this, d_{act} has to be chosen depending on the value of N and the vehicles' physical capabilities like maximum velocity and acceleration.

Step 2 (Concentration Difference): The goal of this stage is to select one single sampling point $(x_s, y_s) \in \mathcal{C}$, that is located as close to the concentration level l_c of interest as possible. Therefore, the absolute value of the concentration differences of the candidate sampling points and the concentration level l_c are compared. Thus, (x_s, y_s) is chosen as

$$(x_s, y_s) = \operatorname*{argmin}_{(x_{c,i}, y_{c,i}) \in \mathcal{C}} |c(x_{c,i}, y_{c,i}) - l_c|. \quad (16)$$

Both steps of the algorithm are depicted in Fig. 2.

In order to overcome situations, where a vehicle is unable to reach its assigned sampling point (e.g. due to motion constraints), each sampling point is replaced by a newly generated one after a specified *lifetime* T_{life}.

As the presented adaptive sampling strategy does not depend on a model of the dispersion process and since it

(a) Set $\mathcal{C}$ of n_c candidate sampling points in $\mathcal{R}_{\mathrm{act}}$.

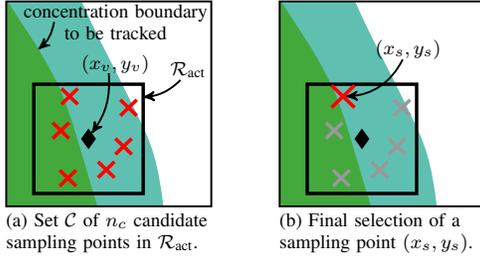(b) Final selection of a sampling point (x_s, y_s).

Fig. 2. Steps of the two-stage adaptive sampling algorithm.

is flexible regarding the number of vehicles, it is easily applicable for different team sizes without need for major plume dispersion process knowledge. Of course, the number of simultaneously trackable boundaries is limited to the number of vehicles.

IV. RESULTS

A. Environmental Setup

An airborne contaminant, disseminated in a work area of $10\mathrm{km} \times 10\mathrm{km} = 100\mathrm{km}^2$, is to be estimated by a team of $n_V = 3$ UAVs. Initially, the UAVs are deployed at positions with a low contaminant concentration. The goal is to track the concentration levels $l_{c,1} = 10^{-7} \frac{\mathrm{kg}}{\mathrm{m}^3}$, $l_{c,2} = 10^{-6} \frac{\mathrm{kg}}{\mathrm{m}^3}$ and $l_{c,3} = 10^{-5} \frac{\mathrm{kg}}{\mathrm{m}^3}$, where each level is assigned to one specific UAV. This setup is depicted in Fig. 1.

B. Vehicle Characteristics

The vehicles' motion dynamics in (2) are approximated by double integrators with maximum velocity $|v_{\mathrm{max}}| = 10\frac{\mathrm{m}}{\mathrm{s}}$ and maximum acceleration $|a_{\mathrm{max}}| = 3\frac{\mathrm{m}}{\mathrm{s}^2}$. The sensory capabilities require each vehicle to remain $N_m = 1\mathrm{s}$ at the sampling location to successfully process a concentration sample. The Kalman filter in the sampling strategy assumes additive Gaussian noise with zero mean and a variance $\sigma^2 = 10^{-4}$. Further, a vehicle's maximum communication range is strongly limited to a distance of 4000m. The grid map representing the work area is divided into cells, where each cell spans a region of $10\mathrm{m} \times 10\mathrm{m}$.

C. Parameters of the Model-Predictive Controller

In order to ensure a reliable communication between all UAVs, the maximum distance is set to a defensive value of $d_{\mathrm{com}} = 3000\mathrm{m}$. The safety distance is set to $d_{\mathrm{col}} = 20\mathrm{m}$ to prevent collisions. Approximation of the vehicles' distances is accomplished according to (3), using $n_\gamma = 8$ inequalities for each pair of UAVs.

The binary variables $b_{\mathrm{com},ij}^{k\gamma}$ and $b_{\mathrm{col},ij}^k$ in (13) are both penalized by the weighting factor $q_\delta = 6000$ whenever they take the value 1 (cf. Section II-C). This way, connectivity is maintained and collisions are avoided. The approximations d_{vs}^k of the distances between the UAVs are weighted by $q_z = 10$. Further, a penalty factor of $q_u = 0.1$ is applied to the control effort $\mathbf{u}^k$.

These parameters along with a prediction horizon of $N = 5\mathrm{s}$ are used in the optimization problem (1). Additionally, the sampling time throughout the simulation is set to $T_s = 1\mathrm{s}$.

Finally, a local MLD subsystem contains a maximum of $\tilde{n}_{V,\mathrm{max}} = 3$ vehicles.

D. Parameters of the Adaptive Sampling Strategy

The proposed adaptive sampling strategy is parametrized by the action and impact range $d_{\mathrm{act}} = d_{\mathrm{imp}} = 90\mathrm{m}$ as well as the number of $n_c = 6$ candidate sampling points preselected in the first step of the algorithm. The maximum lifetime of a sampling point is set to $T_{\mathrm{life}} = 25\mathrm{s}$. The correlation coefficient of the contaminant concentration in adjacent cells within the impact area is set to $r_{\mathcal{I}} = 0.1$.

E. Simulation Results

Simulation results are depicted in Fig. 3 for the initial setup in Fig. 1. For clarity, only each 5th sampling point is plotted in Fig. 3(a)–3(c).

Fig. 3(a) shows that the UAVs quickly locate their assigned concentration levels and subsequently track the respective boundaries. A preliminary linear interpolation of the concentration levels is obtained after 1.5h.

At a simulation time of 4h (cf. Fig. 3(b)), the UAVs are still close to each other due to the communication constraint. At location $(1\mathrm{km}, 6\mathrm{km})$, the generated sampling points deviate from the correct concentration level $l_{c,1} = 10^{-7} \frac{\mathrm{kg}}{\mathrm{m}^3}$ because of the rather small concentration gradient. However, the outermost vehicle successfully relocates its assigned boundary and the cooperative tracking continues. The interpolated concentration distribution already looks similar to the original plume in Fig. 1.

After $T_{\mathrm{sim}} = 6.5\mathrm{h}$, the sampling process is completed as depicted in Fig. 3(c). All vehicles succeeded in tracking their individual boundaries. The motion constraints continuously ensure a reliable communication and collision avoidance throughout the entire simulation.

A final linear interpolation of the concentration samples is shown in Fig. 3(d). The sampled data is compared to the original plume of Fig. 1. It can be seen, that all assigned boundaries have been reconstructed precisely.

V. CONCLUSION

A new sampling strategy for N-boundary tracking of atmospheric plumes by multiple cooperating UAVs was proposed. It effectively integrates concentration estimates and the uncertainty associated with them to determine ideal measurement locations. Simulation results demonstrate that the approach efficiently provides detailed perimeter information in terms of multiple concentration levels of a plume.

Sampling points are generated locally within a vehicle's action range, which represents the area reachable by each vehicle within a fixed time horizon. This implies a tight coupling of the sampling strategy to the employed distributed model-predictive control approach. Based on a MLD model of the multi-vehicle system, efficient motion trajectories meeting collision and communication constraints are obtained and guarantee near optimal vehicle cooperation.

The MPC framework can easily be extended by means of the underlying MLD model and may be coupled to

(a) Sampling points and reconstructed concentration levels after 1.5h.

(b) Sampling points and reconstructed concentration levels after 4h.

(c) Sampling points and reconstructed concentration levels after 6.5h.

(d) Comparison of reconstructed and original concentration levels.
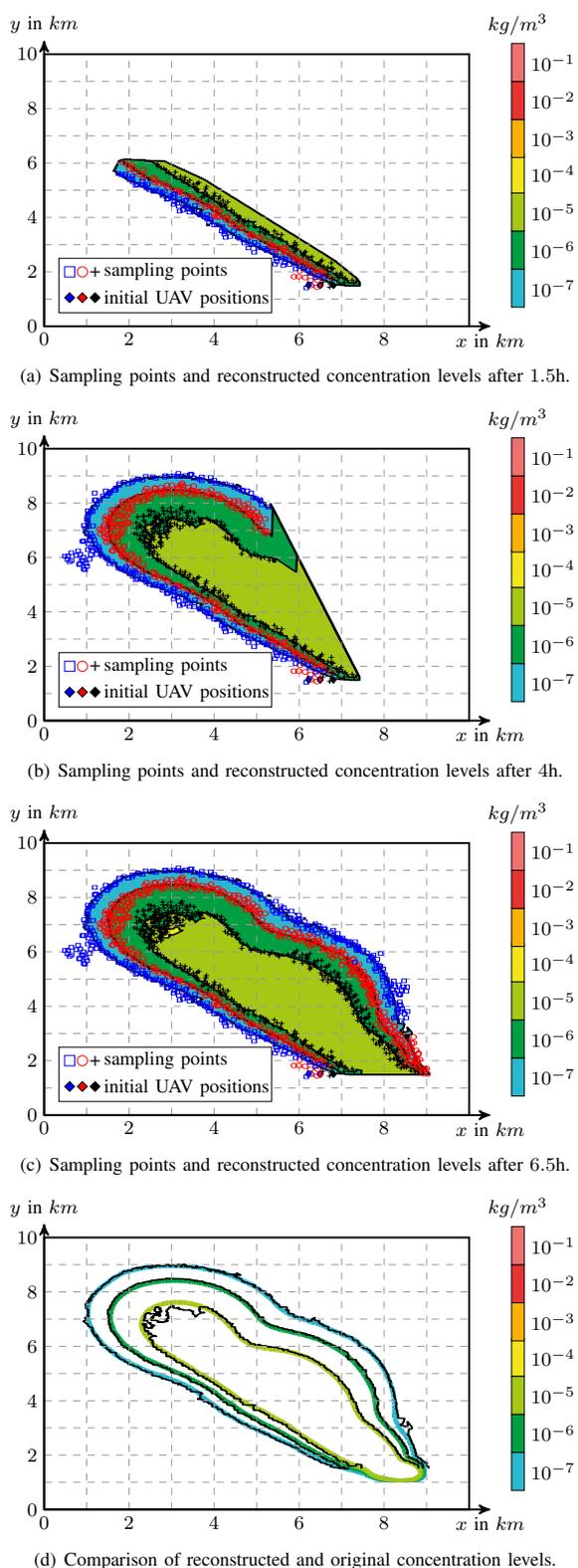
Fig. 3.   Simulation results for the described scenario with 3 UAVs.

different kinds of monitoring strategies. Hence, the presented solution is generalizable to various cooperative multi-vehicle scenarios in environmental monitoring.

REFERENCES

[1] M. Dunbabin and L. Marques, "Robotics for Environmental Monitoring: Significant Advancements and Applications," *IEEE Robotics & Automation Magazine*, vol. 19, no. 1, pp. 24–39, 2012.

[2] Z. Jin and A. L. Bertozzi, "Environmental boundary tracking and estimation using multiple autonomous vehicles," in *Proceedings of the IEEE Conference on Decision and Control*, pp. 4918–4923, 2007.

[3] M. Jaward, D. Bull, and N. Canagarajah, "Sequential Monte Carlo methods for contour tracking of contaminant clouds," *Signal Processing*, vol. 90, pp. 249–260, Jan. 2010.

[4] C. Hsieh, D. Marthaler, B. Nguyen, D. Tung, A. Bertozzi, and R. Murray, "Experimental validation of an algorithm for cooperative boundary tracking," in *Proceedings of the American Control Conference*, pp. 1078–1083, 2005.

[5] F. Zhang and N. E. Leonard, "Generating contour plots using multiple sensor platforms," in *Proceedings of the IEEE Swarm Intelligence Symposium*, pp. 309–316, 2005.

[6] P. P. Neumann, S. Asadi, A. J. Lilienthal, M. Bartholmai, and J. H. Schiller, "Autonomous Gas-Sensitive Microdrone: Wind Vector Estimation and Gas Distribution Mapping," *IEEE Robotics & Automation Magazine*, vol. 19, no. 1, pp. 50–61, 2012.

[7] C. J. Cannell, A. S. Gadre, and D. J. Stilwell, "Boundary Tracking and Rapid Mapping of A Thermal Plume Using an Autonomous Vehicle," in *OCEANS 2006*, pp. 1–6, 2006.

[8] S. Susca, S. Martinez, and F. Bullo, "Monitoring environmental boundaries with a robotic sensor network," in *Proceedings of the American Control Conference*, 2006.

[9] M. Boardman, J. Edmonds, K. Francis, and C. M. Clark, "Multi-robot boundary tracking with phase and workload balancing," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 3321–3326, Oct. 2010.

[10] R. Graham and J. Cortés, "Cooperative adaptive sampling via approximate entropy maximization," in *Proceedings of the IEEE Conference on Decision and Control*, 2009.

[11] P. Kathirgamanathan, *Source Parameter Estimation of Atmospheric Pollution from Accidental Releases of Gas*. PhD thesis, Massey University, 2003.

[12] J. M. Stockie, "The Mathematics of Atmospheric Dispersion Modeling," *SIAM Review*, vol. 53, no. 2, pp. 349–372, 2011.

[13] V. N. Christopoulos and S. Roumeliotis, "Adaptive Sensing for Instantaneous Gas Release Parameter Estimation," in *Proceedings of the IEEE International Conference on Robotics and Automation*, pp. 4450–4456, 2005.

[14] P. Kathirgamanathan and R. McKibbin, "Inverse Modelling for Identifying the Origin and Release rate of Atmospheric Pollution-An Optimisation Approach," *Proc. MODSIM*, 2003.

[15] L. Rossi, B. Krishnamachari, and C.-C. Kuo, "Distributed parameter estimation for monitoring diffusion phenomena using physical models," in *IEEE Conference on Sensor and Ad Hoc Communications and Networks*, pp. 460–469, 2004.

[16] C. J. Cannell and D. J. Stilwell, "A comparison of two approaches for adaptive sampling of environmental processes using autonomous underwater vehicles," in *Proceedings of the MTS/IEEE OCEANS Conference*, pp. 1514 –1521 Vol. 2, 2005.

[17] C. Reinl, *Trajectory and Task Planning for Cooperating Vehicles: Discrete-Continuous Modeling and Optimization*. PhD thesis, Technische Universität Darmstadt, 2010.

[18] J. Kuhn, C. Reinl, and O. von Stryk, "Predictive Control for Multi-Robot Observation of Multiple Moving Targets Based on Discrete-Continuous Linear Models," in *Proceedings of the 18th IFAC World Congress*, pp. 257–262, 2011.

[19] Sage Management, "Second-order Closure Integrated Puff," 2000.

[20] A. Bemporad and M. Morari, "Control of systems integrating logic, dynamics, and constraints," *Automatica*, vol. 35, pp. 407–427, 1999.

[21] H. P. Williams and S. C. Brailsford, "Computational logic and integer programming," in *Advances in linear and integer programming* (J. E. Beasley, ed.), pp. 249–281, Oxford University Press, Inc., 1996.