J Intell Robot Syst
DOI 10.1007/s10846-016-0338-z

Swarm Distribution and Deployment for Cooperative Surveillance by Micro-Aerial Vehicles

Martin Saska · Vojtěch Vonásek · Jan Chudoba · Justin Thomas · Giuseppe Loianno · Vijay Kumar

Received: 15 December 2014 / Accepted: 11 January 2016

Abstract The task of cooperative surveillance of preselected Areas of Interest (AoI) in outdoor environments by groups of closely cooperating Micro Aerial Vehicles (MAVs) is tackled in this paper. In the cooperative surveillance mission, finding distributions of the MAVs in the environment to properly cover the AoIs and finding feasible trajectories to reach the obtained surveillance locations from the initial depot are crucial tasks that have to be fulfilled. In addition, motion constraints of the employed MAVs, environment constraints (e.g. non-fly zones), and constraints imposed by localization of members of the groups need to be satisfied in the planning process. We formulate the task of cooperative surveillance as a single high-dimensional optimization problem to be able to integrate all these requirements. Due to numerous constraints that have to be satisfied, we propose to solve the problem using an evolutionary-based optimization technique. An important aspect of the proposed method is that the cooperating MAVs are localized relatively to each other, rather than using a global localization system. This increases robustness of the system and its deploy-ability in scenarios, in which compact shapes of the MAV group with short relative distances are required.

M. Saska (✉) · V. Vonásek · J. Chudoba
Department of Cybernetics, Faculty of Electrical Engineering, Czech Technical University in Prague, Technicka 2, 16627, Prague, Czech Republic
e-mail: martin.saska@fel.cvut.cz

V. Vonásek
e-mail: vonasek@labe.felk.cvut.cz

J. Chudoba
e-mail: chudoba@labe.felk.cvut.cz

J. Thomas · G. Loianno · V. Kumar
GRASP Lab, University of Pennsylvania, 3330 Walnut Street, Philadelphia, PA 19104, USA

J. Thomas
e-mail: jut@seas.upenn.edu

G. Loianno
e-mail: loiannog@seas.upenn.edu

V. Kumar
e-mail: kumar@seas.upenn.edu

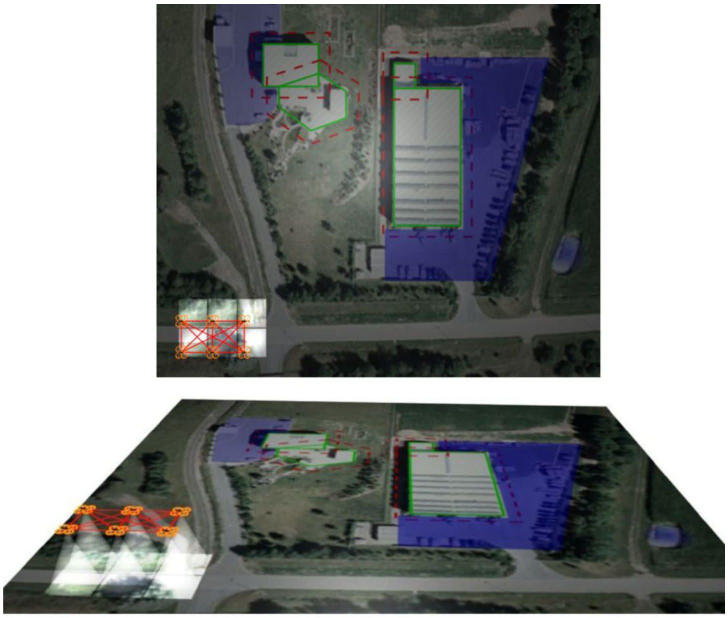
Keywords Micro-aerial vehicles · Cooperative surveillance · Swarms · PSO · Visual localization · Motion planning · Swarm coverage

1 Introduction

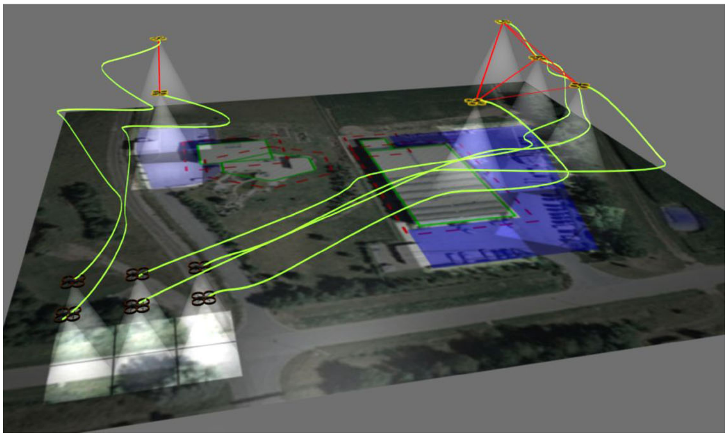
The task of cooperative surveillance in outdoor areas by a fleet of relatively stabilized Micro Aerial Vehicles (MAVs) is investigated in this paper. In the proposed scenario (see Fig. 1 for motivation), identical MAVs (unmanned quadrocopters) are aimed at cooperative observation of a set of Areas of Interest (AoI) assigned by a human operator (a security expert) in an environment with obstacles. The obstacles may be physical obstacles such as buildings, or non-fly areas in which the MAVs are not supposed to fly. The surveillance of

Published online: 10 February 2016

Fig. 1 Motivation of the investigated swarm deployment task



(a) Initial position of the MAV group that is prepared for the surveillance mission (top view and side view). Obstacles are denoted by the green lines, borders of non-fly zones (dilated obstacles) by the dashed red lines and areas of interest are depicted as the blue regions.



(b) MAV group deployed in the environment for covering the areas of interest by onboard surveillance cameras. The range of the cameras is visualized by the white pyramids. The yellow curves represent trajectories from the initial position (the depot) into the final location and the red lines represent the relative localization linkages.

an AoI (or its part) can be realized by a single MAV flying at a sufficiently low altitude (the altitude from which the area is observed with an optimal resolution) or by several MAVs flying in a higher altitude with overlapping views of their surveillance cameras. It is assumed that the utilization of several MAVs observing the same point in the environment from different angles brings an added value for the mission. Such cooperative surveillance is necessary for covering complex areas of interest, which cannot be sufficiently observed by a single MAV. Moreover, the cooperative surveillance increases robustness of the system, as the mission operator can still obtain data from surveillance sensors in a case of a single MAV failure.

The deployment of closely operating groups of unmanned MAVs requires a precise localization of the individual MAVs for which precision of open access systems (such as GPS) is not sufficient. The particular team members are required to fly in relative distances, which may be significantly smaller than the precision of GPS. Moreover, GPS lacks sufficient reliability in urban areas, whose surveillance is one of our main target scenarios. To replace such global localization system with insufficient precision and reliability, an onboard localization of neighbours in the swarm needs to be employed, which brings additional constraints that have to be satisfied in motion planning and swarm stabilization. The multi-MAV surveillance approach presented in this paper is designed to satisfy these constraints.

As an example of the onboard positioning system, we rely on a relative localization system [1] that is realized by monocular cameras and simple localization patterns attached to MAVs in experimental evaluation of the method. This system provides sufficient precision, reliability, and fast response to provide feedback for the mutual stabilization of a compact group of MAVs, if the group members respect a set of localization constraints. The constraints are specified mainly by image resolution, the viewing angle of cameras, and by the size of the localization pattern. This results in constraints on relative positions of neighbouring MAVs in the group that can be precisely defined prior the mission, and thus integrated into the planning approaches. Details on the allowed mutual position of the camera and the pattern if using the system of visual relative localization may be found in [2]. The localization system requires that the neighbouring MAVs within the group are kept in the constrained relative positions (later called as the relative localization constraints) during the entire mission to be able to track them and to ensure mutual collision avoidance. Therefore, these localization constraints need to be integrated into the proposed method designed for the deployment of MAV groups into locations that are appropriate for the surveillance.

Since the MAV control system uses only a limited information on positions of neighbours and all MAVs of the fleet are assumed to be identical, let us call the employed MAV group as a "swarm" to highlight that MAVs may be mutually replaced. This enables us to utilize a localization system without any need to distinguish the group members, which significantly reduces computational requirements as it is discussed in [2]. Such localization can be then realized fast and run on-board, and may be used in fast position feedback for the swarm stabilization. Let us call the set of static positions of all MAVs, which are connected into a compact group via the relative localization linkages, a *swarm distribution*. The task of the motion of MAVs from the initial depot into the static swarm distribution (a set of feasible trajectories) is called *swarm deployment* in this paper.

In the proposed swarm deployment method, a single optimization process is used to find a final swarm distribution in the environment together with feasible trajectories of all MAVs to reach these positions from the depot. This ensures that the surveillance locations are reachable by MAVs keeping the motion and relative localization constraints.

Since the quality of coverage of the AoIs depends also on the altitude of MAVs, the optimization vector representing the entire solution consists of the 3D positions of all MAVs. The dimension of the optimization vector is therefore $3n_r$, where n_r is the number of employed MAVs. We propose to solve this high dimensional optimization problem by the Particle Swarm Optimization (PSO) technique due to its evolutionary character. The key idea of the proposed approach is to consider the optimization vector as a swarm of real MAVs flying in the space of possible swarm distributions. During the evolution, the motion and relative localization constraints are tested in each transition between PSO populations as described in details in Section 5. The feasible trajectories of all MAVs can be then extracted directly from the progress of evolution of the best particle in the PSO population. Although the initial plan of the swarm deployment is obtained in a centralized way, the MAV motion into the obtained swarm distribution for the surveillance may be realized using local controllers and information from onboard sensors only.

This paper is an extension of the conference paper [3]. In [3], a basic description of the method solving the *swarm distribution* and *swarm deployment* tasks is provided with focus on extensive simulations and tuning of parameters of the employed PSO technique. In addition, experiments with a fleet of real MAVs are presented in [3], aimed at verifying the possibility of using the relative visual localization onboard of flying robots with sufficient precision and reliability. The performance of the planning was evaluated by Vicon motion capture system being used as a ground truth reference of the MAV localization. In the experiments, it was shown that the visual relative localization system may be employed onboard of MAVs, and that the proposed method for solving the *swarm distribution* and *swarm deployment* tasks provides feasible solutions respecting the localization and motion constraints. Nevertheless, the experiments presented in [3] rely on the Vicon system in control feedback and the relative visual localisation is only passively used for later evaluation of its performance.

Here, we present more detailed description of the proposed method with both, theoretical and experimental analyses of performance of the complex system. The comprehensive state-of-the-art description enables to clearly specify contribution of the method relatively to existing multi-robot literature. In addition to [3], we also provide a description of the utilized visual relative localization technique, which is important to understand the overall system and mainly the localization and motion constraints being the core of the proposed planning approach. Finally, we present new MAV stabilization and control method suited for real-world deployment (i.e., without a motion capture system) of the proposed system. New outdoor and indoor experiments show performance of the complex system relying strictly on onboard sensors, and in the same way as it will be done in the target scenarios.

1.1 State of the Art and Progress Beyond State of the Art

The main topics being investigated in multi-MAV systems nowadays cover problems with maintenance of communication in large groups [4, 5], allocation of various simultaneously solved tasks [6, 7], studies of the swarm behavior by understanding behaviors of swarm members [8–10], and stabilization of swarms with possibility of avoidance of obstacles in near proximity [11–13]. This paper investigates mainly the problem of swarm stabilization in environment with obstacles and partly also autonomous distribution of tasks (observation of particular AoIs) within the group. Similar approaches for control of swarms of unmanned aerial vehicles can be found in [14–16]. A survey of strategies for distribution of various multi-robot tasks within team of MAVs is available in [17].

In this paper, we investigate problems of swarm stabilization constrained by limited relative localization. In some aspects, this problem may be analogical to approaches solving swarm control with limited communication requirements (see e.g. [18] and [19]). In [18], the requirements on the swarm stability are specified by a direct graph topology. In [19], a Lyapunov's theory is used for analysis of convergence of the system.

From the above mentioned, the most related work is the paper [19], where swarming of UGVs is solved. We are also aimed at developing a system for stabilization of multi-MAV system in desired shapes while keeping small distances between robots. Beyond [19], MAV swarm principles are investigated in 3D here, and the designed rules for swarm control are suited for the requirements of visual relative localization.

The tasks of *swarm distribution* and *swarm deployment* in cooperative surveillance are the most related to the research of multi-robot coverage that was firstly studied for UGVs (see e.g. works [20, 21] and references reported therein). The multi-robot coverage in the task of surveillance is investigated in [22], where a group of mobile robots is controlled with GPS in feedback. Biologically inspired solutions of swarm coverage can be found in [23, 24].

For MAVs, a continuous coverage problem is usually solved, in which MAVs repeatedly follow a surveillance trajectory (see [25, 26]). The static cooperative coverage by MAVs hovering in static positions can be found in [27, 28]. The work in [27] is the most relevant to our approach. It uses a distributed method for stabilization of MAVs in fixed positions that are designed for monitoring of given areas by multiple downward facing cameras. The optimization criterion for the multi-MAV distribution problem is also the gained information per pixel as it is proposed in our method. Nevertheless, the algorithm in [27] relies on precise motion capture system (Vicon) for the global localization. The surveillance coverage by MAVs in areas without GPS is investigated in [28]. Our method goes beyond this work, since the approach in [28] does not enable stabilization of MAVs observing the same area (swarm with small relative distance). The work in [28] solves mainly a visual-SLAM with the aim maximizing relative distances between the team members.

Finally, let us refer to some examples of evolution based optimization methods that are used in robotics for trajectory planning [29, 30], optimal control [31], and locomotion generation [32]. All these methods are trying to reach a desired behavior whose efficiency is described by a fitness function. This function is optimized in a search space of parameters (parameters of the trajectory in the trajectory planning, parameters of controllers, joint angles in modular robots, etc.). In our method, positions of all MAVs in the swarm are encoded into a unique solution (optimization vector) of the evolutionary optimization, which is a revolutionary approach to MAV swarm control. The efficiency of positions of all MAVs (the swarm shape and its position) with respect to the cooperative surveillance is considered in the proposed fitness function. Moreover, the evolutionary based approach enables to simply integrate various swarm and motion constraints into the optimization.

The main contributions of this paper are the following:

- It presents a novel approach to multi-MAV cooperative surveillance, which is suited for outdoor deployment of closely cooperating MAVs flying with small relative distances.
- It provides a study of performance of the evolution based optimization method that uses tangible particles with motion constraints enforced by deployment of MAVs in a real world scenario.
- It shows an advantage of using a motion planning technique integrated in the core of evolution based optimization methods.

2 Problem Statement and Preliminaries

In the investigated scenario of autonomous cooperative surveillance, a set of AoI is given to a limited number n_r of autonomous robots (MAVs), aimed at finding a static swarm distribution to cover these areas (see Fig. 1 for an overview of the mission scenario). In addition, a set of no-fly zones, considered as obstacles for MAVs, is known prior to the mission. The areas of interest are gathered into an *AoI Map* and the workspace of robots with the no-fly zones is described in an *Environment Map*, as shown in Section 4.1. During the autonomous swarm deployment into the proper MAVs locations, the system has to respect: 1) motion constraints of particular MAVs defined by their motion capabilities and utilized controller (see Section 6 for details), 2) localization constraints that depend on the localization cameras and the localization system itself (see Section 3 for details), and 3) sensing capabilities of the surveillance cameras (based on the viewing angle and resolution). All these parameters are also known prior to the mission and have to be integrated into the final plans during the planning of the *swarm distribution* and *swarm deployment* tasks. The constraints imposed by the swarm stabilization based on the relative localization have to be satisfied in both the final static swarm distribution at the surveillance locations as well as during the swarm deployment from its initial location (a depot).

In case of an insufficient size of the swarm, which is not capable to cover the given set of locations of interest completely by its limited sensors and due to a limited altitude, the set of AoI is covered to maximize the information collected together by the MAV group. Each part of the set of AoIs can be then observed either by one MAV flying at the altitude that is specified as the optimal altitude for the surveillance, or by several MAVs flying at higher altitudes that are observing the same area. The value of cost function, which indicates quality of the swarm distribution, depends on the overall amount of information gained together by all MAVs (details on the cost function implementation are presented in Section 4.1). The limited altitude of MAVs is considered in our approach from two reasons. The altitude for operation of MAVs is always limited by safety regulations, and it may vary from meters in densely populated urban areas to hundreds meters in countryside. The second reason is that the resolution of cameras is insufficient for the purpose of the surveillance task above a threshold altitude.

3 Visual Relative Localization

The core of the proposed approach that enables the swarm deployment in the surveillance application is an onboard localization of MAVs of the group. To demonstrate the performance of the system in real world conditions, we rely on visual relative localization of neighboring vehicles in this paper. This makes possible stabilization of the swarm of MAVs in relative positions that are defined by the planning system. In particular, we rely on a vision technique [1] based on a detection of black and white (B/W) pattern, which was originally developed within our group for stabilization heterogeneous MAV-UGV formations of fixed shapes [33–35]. In verification experiment presented in this paper, the vision based stabilization is used as an embedded solution, which is implemented on a Gumstix Overo computer with attached Caspa camera (see [2] for details on the HW solution). The employed camera module satisfies requirements of robotic swarms being deployed in outdoor environments and it provides precision in order of units of centimeters for measuring distances in units of meters with 50Hz update frequency. This performance is sufficient for the presented experiments verifying the proposed planning method. The fast update rate allows us to directly integrate the vision based relative localization in the feedback loop of swarm control and its shape stabilization, or to fuse data obtained from the relative localization system with additional sensors for increasing the swarm stability (see Section 6 for the scheme of the employed controller).

Using a module with simple monocular camera introduces operational constraints of the relative localization that were analytically and experimentally identified in [2]. These constraints specify relative positions in which neighbouring vehicles in the swarm equipped with the identification pattern may be localized with sufficient precision and reliability. It influences allowed shape of the swarm and specifies localization constraints that have to be kept during the swarm deployment. This knowledge needs to be considered in the proposed planning algorithm for the *swarm distribution* and *swarm deployment* tasks. Proper consideration of the employed visual relative localization decreases the overall uncertainty in positioning of robots within the swarm and increases reliability of the overall system.

The requirements on fast localization, necessary for its using in control feedback, and onboard usability require low computational demands for the image processing algorithm. Therefore, simple circular patterns composed of concentric black and white circles of known diameter are used for rapid detection and identification of necessary parameters, i.e., positions of circles in the image and their distance from the localization camera.

A fast flood-fill method is used for searching the image for continuous areas of pixels that are tested whether they represent the searched pattern, using statistical information gathered on-the-fly. The searching process can be initiated from any position in the image, which enables employing pattern tracking and scanning only the area surrounding expected position of the pattern. In the beginning of the image processing, a continuous segment of black pixels is found and tested for roundness and minimum size (see [1] for implementation of these tests). In the center of a detected black circle, the flood-fill method is restarted for searching for a white area and the test for roundness and minimum size is repeated. After checking ratio of both areas and its concentricity, ellipse center, covariance matrix and ellipse (co-)vertices of the found segment are calculated and this information is used for obtaining size of the pattern, and so its relative distance. Then, the vertices of the ellipse in image coordinates are transformed into canonical camera coordinates. The camera length, optical center and its radial distortion are taken into account in this transformation. The transformed vertices are used for precise calculation of the centre and axes of the ellipse in the canonical camera form for obtaining the relative position of the pattern.

4 Swarm Motion Planning Problem

The swarm motion planning problem solved in this paper consists of two main challenges: 1) how to find a suitable swarm distribution that covers all AoIs, and 2) how to to find feasible trajectories of MAVs (the swarm deployment) from a depot to the final positions, which are described by the swarm distribution. In both cases, linkages between MAVs required by the relative localization need to be satisfied. In addition, the motion constraints and dynamics of MAVs need to be considered in the obtained trajectories. Due to the numerous constraints, it is computational demanding to find a solution that consists of both the optimal swarm distribution together with the optimal swarm deployment.

Therefore, we propose to optimize the swarm distribution to maximize the coverage of desired AoIs, but we do not aim at finding optimal trajectories to reach this swarm distribution. Nevertheless, the proposed method guarantees the feasibility of the proposed trajectories, which is achieved by tracking of feasible motion of MAVs during the evolution of the optimization process. The optimization process starts from a solution, where all MAVs are located in the depot (the initial positions of MAVs), to ensure feasibility of trajectories from their beginning. Always if a new swarm distribution is found during the optimization process (in each step of the evolution of the optimization algorithm), a motion planning technique is used to reach this new solution in a simulation ensuring feasibility of the trajectories. If the feasibility would be violated (some of the desired requirements may not be kept any more), the motion simulation is interrupted, and the optimization process is asked for a new solution to follow. The cost function used in the optimization process is described in the next subsection, which is followed by a description of the employed optimization algorithm.

4.1 Cost Function

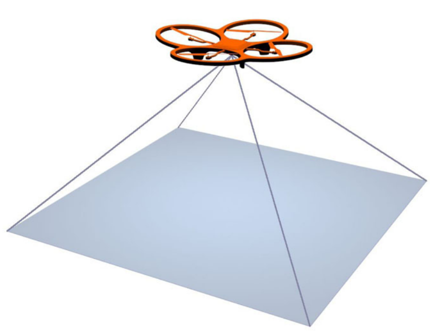
The cost function describes the quality of coverage of AoIs achieved by a swarm distribution X, which is represented as a vector of 3D positions of n_r MAVs. The environment is represented by a matrix $A \in \mathbb{R}^{a \times b}$ with values of its elements between 0 and a constant A_{max}. We refer to A as *AoI map* in the rest of the paper. An example of construction of the AoI map is shown in Fig. 3. The cells designed as AoIs with maximal priority of surveillance are filled with maximum value $A_{x,y} = A_{max}$, while cells representing regions that do not have to be observed by MAVs are filled with zeros. If required, any value between 0 and A_{max} can be applied for some of AoIs regions if a lower priority of observing the particular region needs to be included. The meaning of the lower number in the particular cell may be also interpreted in a way that this area may be observed from a higher altitude by one MAV to provide sufficient information. The resolution of A, which is determined by constants a and b, depends on the application. As usual, if the shape of AoI does not need to be determined precisely, a sparse grid with smaller constants a and b can be used.

The quality of coverage of desired AoIs by a set of MAVs is evaluated by the cost function

$$CF(X) = \sum_{x=1}^{a} \sum_{y=1}^{b} \max \left(0,\, A_{x,y} - \sum_{i \in O(x,y)} \frac{S_{opt}}{S_i} A_{max} \right), \tag{1}$$

where S_i is area of the rectangle representing the part of the workspace that can be observed by the i-th MAV (see Fig. 2a for visualization of the model of the surveillance camera). The size and position of the rectangle depend on the viewing angle of the employed sensor, actual altitude of the i-th MAV and its Pitch and Roll angles. S_{opt} is area of the region, which is observed by MAV flying at the altitude determined as the "optimal" altitude based on the particular application (this altitude is usually specified by the security expert). We assume that MAV flying at the "optimal" altitude provides a sufficient amount of information on the part of AoI assigned with the A_{max} value, which is covered by its surveillance sensor. An MAV at a lower

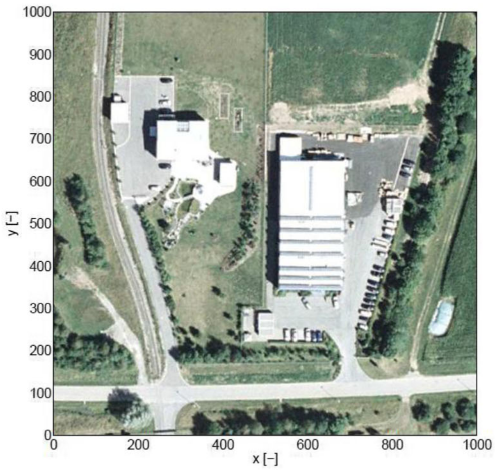
Fig. 2 Models used in the PSO planning loop



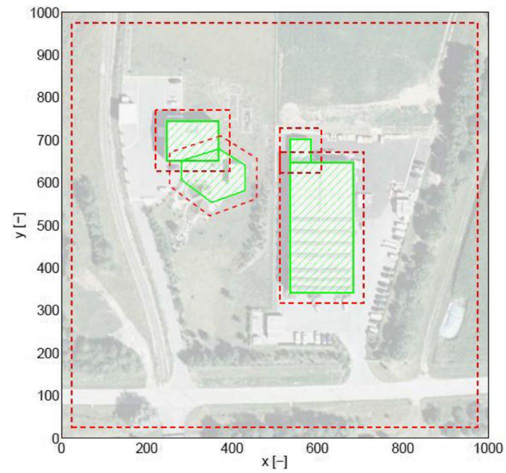
(a) Model of the surveillance camera used in the cost function of the swarm motion planning.

(b) MAV model employed in the collision avoidance.

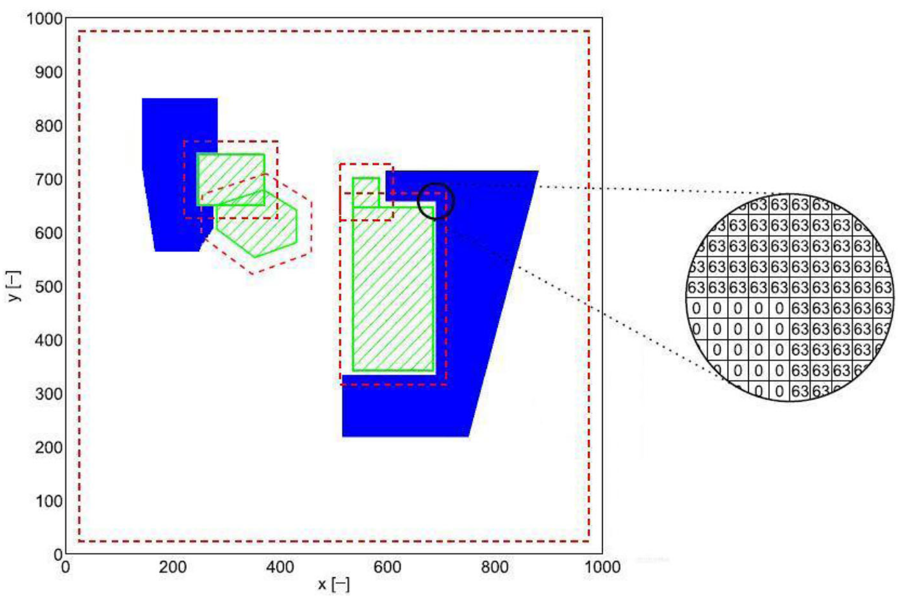
altitude does not gain more information per square unit. On the contrary, we suppose that the information per square unit decreases linearly with increasing area S_i (the total amount of gained information from the sensor is constant). The set $O(x, y)$ consists of MAVs that fully observe the particular cell $A_{x,y}$ by their surveillance sensors. The required information from the element $A_{x,y}$ can be then obtained by a single MAV flying at the "optimal" altitude, or cooperatively by a set of MAVs flying at higher altitudes. The sum of remainders (after subtraction of contributions of all MAVs) of the initial values assigned to each cell $A_{x,y}$ is considered as the cost value of the particular solution (Fig. 3).



(a) Satellite map of an industrial area that has to be under surveillance of MAVs.



(b) Highlighted obstacles and borders of dilated no-fly zones.



(c) Final map with depicted areas of interest.

Fig. 3 An example of construction of a map with no-fly zones (obstacles) and areas of interest

Our intent was to integrate into the cost function preliminary experimental results with a fleet of MAVs observing the same area from slightly different locations and therefore also different observation angles. It was found out that it may often happen that an object being simply detected in an image from one MAV is not visible in images taken from another MAV flying in slightly different location. The reason can be the slightly different observation angle, usually different setting of auto exposition, occlusion with obstacles, shadows, etc. Moreover, as mentioned, an optimal altitude in which the observed object is identified in particular surveillance applications can be usually simply defined, and it has no sense to fly below this level. In the presented cost function, these observations are integrated together with possibility to prioritize areas with higher importance for the surveillance task. Nevertheless, this cost function is only an example and it can be simply changed for different surveillance scenarios.

4.2 Particle Swarm Optimization

To solve the swarm distribution problem in Eq. 1, $3n_r$ variables need to be optimized at once, as each MAV is represented by its 3D position in the environment and n_r MAVs belong to the swarm. The cost function, that is described above may have, depending on the number of AoIs and their distribution in the environment, several local optima. Such high dimensional optimization problem with plenty of local extremes can be effectively solved using evolutionary-based optimization methods. In addition, the evolutionary process is especially appealing for the proposed approach, in which each state of the swarm distribution is considered as one solution of the optimization and the history of evolution of this solution is considered as the solution of the swarm deployment.

From the available population based optimization methods, the PSO algorithm [36] was chosen as the best candidate for this study. The PSO method takes advantage of the swarm intelligence for increasing convergence of the optimization and for reduction probability of getting stuck in a local optima. In PSO, each solution (called particle) of the optimization problem represents a point in a multidimensional space of solutions. The group of particles is often called “PSO swarm”, but we will rather use the term population, since the group of relatively stabilized MAVs is denoted as the swarm in this paper. The essence of the PSO method is inspired by a social behavior of birds and fish (by movement of their flocks and schools). Therefore, it offers the possibility of considering the movement of swarms of MAVs analogically as the movement of particles in PSO.

In the PSO optimization, movement of each particle is influenced by its own experience and also by a social knowledge shared amongst the particles. Let b_j denote position of particle j, where the particle achieved best value of the cost function so far. This position represents the particle’s own experience. Let b_g denote the best position amongst all particles, which represents the social knowledge. The global best particle b_g also represents the actual solution of the problem being solved. In the PSO algorithm, each particle j is represented by its D-dimensional position $p_j(t)$ and its current velocity vector $u_j(t)$. The dimension of the searching space D is equal to $3n_r$. The velocity vectors of all particles are updated by the following rule in each iteration t:

$$\begin{aligned} u_j(t) = u_j(t-1) &+ \Phi_1(b_j - p_j(t-1)) \\ &+ \Phi_2(b_g - p_j(t-1)), \end{aligned} \tag{2}$$

where Φ_1 and Φ_2 are obtained as

$$\Phi_k = c_k \begin{pmatrix} ran_{k,1} & 0 & 0 \\ 0 & \ddots & 0 \\ 0 & 0 & ran_{k,D} \end{pmatrix}, k \in \{1, 2\}. \tag{3}$$

The constants c_1 and c_2 weight the particles’ own experience and the social knowledge. Random numbers $ran_{1,1}, \cdots, ran_{1,D}$ and $ran_{2,1}, \cdots, ran_{2,D}$ are obtained with a uniform distribution between 0 and 1.

In the second step of the PSO iteration, the positions of all particles are updated as

$$p_j(t) = p_j(t-1) + u_j(t). \tag{4}$$

The velocity and position update rules are applied repeatedly until t_{max} iterations is achieved. Another stopping criterion may depend on quality of the obtained solution, which can be simply evaluated using the cost function (1). In addition, the possibility of evaluation of the obtained result prior the mission is beneficial in safety critical applications. The feasibility of the solution is always guaranteed due to the *Feasibility check* module that is employed in each step of the optimization, which also increases robustness of the swarm deployment task. The disadvantage of the employed PSO method is missing evidence of convergence into a satisfactory solution and dependence on its initialization. A commonly used strategy for increasing robustness of the planning process is multiple re-initialization of the optimization problem and selecting the best solution achieved in these trails.

Two commonly used extensions of the velocity update rule were successfully tested in the analyses of the algorithm performance (see our previous publication [3]): 1) A limit on velocity of particles, which is applied if any component of the result of the velocity update rule $u_j(t)$ exceeds a given constant u_{max}. Then the exceeding vector components are replaced by u_{max}. 2) A linear decrease of the inertia of particles from an initial value w_{start} to a value w_{end}, $w_{start} > w_{end}$, at the end of the PSO process. It ensures that the influence of the velocity vector from the previous iteration decreases during the optimization process. The stored velocity from the previous iteration is then updated before the new calculation of the velocity update rule as

$$u_j(t-1) := \left(w_{start} + t\frac{w_{end} - w_{start}}{t_{max}}\right) u_j(t-1). \quad (5)$$

For employing the PSO method in the swarm distribution problem, each particle encodes 3D positions of all n_r MAVs (i.e., swarm distribution X): $X = (x_1, y_1, z_1, \ldots, x_{n_r}, y_{n_r}, z_{n_r})$. The quality of the particles can be directly evaluated by the cost function in Eq. 1. The scheme of the encoding of swarm of MAVs in PSO particles is depicted in Fig. 4.

5 Tangible Particle Swarm Optimization

A solution of the swarm distribution problem is obtained using the above described PSO optimization, where PSO particles encode 3D positions of the MAVs in the environment. The motion of these particles in the solution space is described by Eq. 4. Due to swarm constraints, that are not considered in this equation, this simple PSO cannot ensure that new positions of MAVs described by particle $p_j(t)$ are reachable from the previous position $p_j(t-1)$. Therefore, we propose a novel extension to PSO optimization, called Tangible PSO, that can solve the swarm deployment and swarm distribution problems simoultaneously. The output of the Tangible PSO algorithm is a feasible plan of the MAVs movement from the initial configuration to the final swarm distribution.

In comparison with the standard PSO, where the velocity and position update rules are simply applied in each PSO iteration, the proposed Tangible PSO verifies the transition from $p_j(t-1)$ to $p_j(t)$ using a simulation of the MAV swarm motion to ensure reachability of the new solution. A simple scheme of such an approach is shown in Fig. 6. In each iteration, new positions $p_j(t)$ of PSO particles (and therefore positions of virtual swarms) are found based on the PSO update rule in Eq. 4. Mutual collisions in the proposed positions of MAVs are found in all virtual swarms separately. If the i-th MAV would collide in its new position $p_j(t)$ with another MAV, its position is set to $p'_j(t)$ that is in a sufficient distance from the position where the collision is detected. The position $p_j(t)$ is changed to $p'_j(t)$ also if there is a collision with an

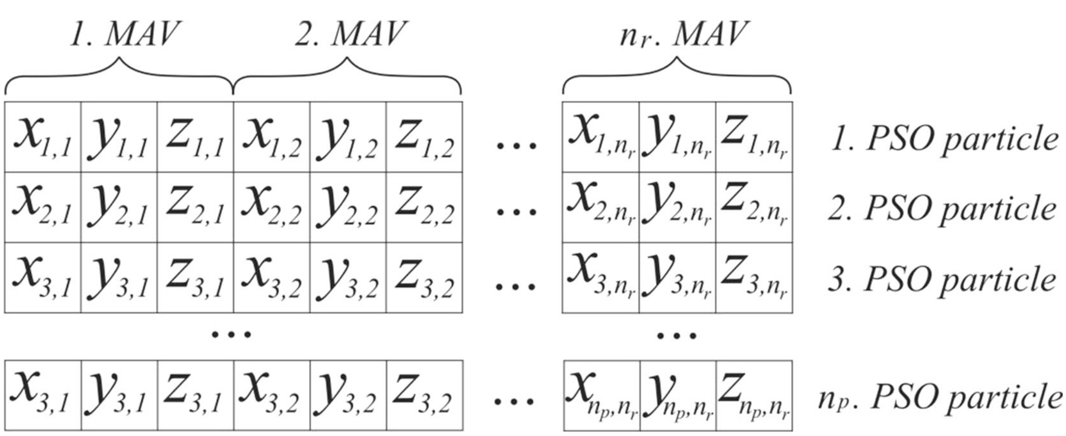


Fig. 4 Scheme of the PSO population

obstacle or if $p_j(t)$ lies behind the operational area. This is depicted in Fig. 5a.

The model composed from two cones shown in Fig. 2b is used to represent the MAVs for the collision detection. Such a simplified description of the quadrotor involves its physical shape and also the zones where the neighbouring MAVs would be affected by airflow from propellers.

In the following step, a motion planning algorithm is used to verify reachability between the swarm positions $p_j(t-1)$ and new positions $p'_j(t)$. Motion and swarm constraints have to be considered during the motion planning to ensure that the transition from $p_j(t-1)$ to $p'_j(t)$ is feasible. The motion planning has to be computed as fast as possible, because it is run in each iteration of the Tanglible PSO and for each MAV of each virtual swarm. The transition from $p_j(t-1)$ to $p'_j(t)$ can be computed either using combination of path planning and trajectory following or using single motion planning technique, which can provide also trajectories for individual MAVs.

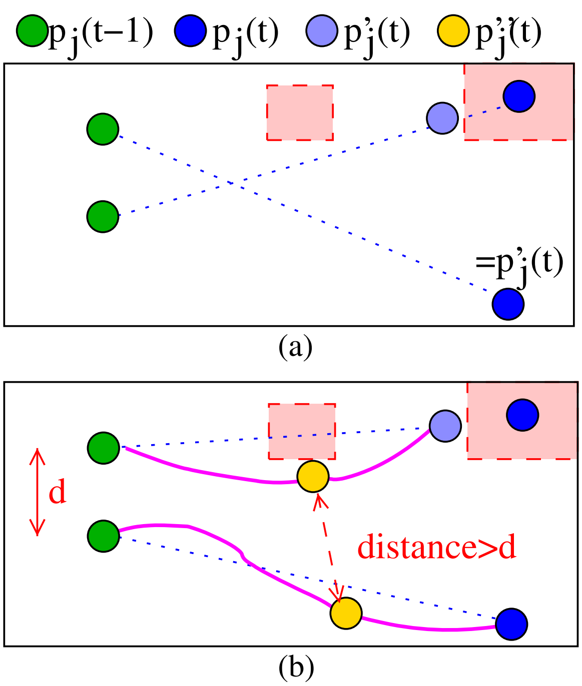


Fig. 5 Example of computation of new positions $p_j(t)$ of a swarm of two MAVs. The new positions $p_j(t)$ of each MAV (*blue circles*) are suggested from PSO algorithm (**a**). As one of the position lies in the obstacle (*top right corner*), it is shift to position $p'_j(t)$. After a path plan is computed, a mutual collision between both MAVs is detected, which is resolved by switching their goal positions (**b**). The trajectory following is then run from $p_j(t-1)$ to $p'_j(t)$ but the constraints are violed at point $p''_j(t)$ (*yellow*). The first (*top*) helicopter has to stop at $p''_j(t)$ due to the obstacle and the second one (*bottom*) cannot continue the flight much because it has to keep the relative distance d

In the former case, a path planning method provides collision-free 3D paths in the environment for each MAV, and a path following method is then used to compute trajectory for MAV along the path considering motion and swarm constraints. For fast computation of 3D paths, Visibility graph (VG) approach [37] can be used. The VG approach provides paths for each MAV separately, which can lead to mutual collisions. Therefore, the paths need to be checked before they are passed to the trajectory following procedure. Possible mutual collisions between paths are identified for each of the MAVs. If a collision between two MAVs is identified, the target locations of these MAVs are switched (Fig. 5b). This simple but efficient approach is allowed since the MAVs in the swarm are considered as equivalent entities (Fig. 6).

The efficiency of this simple procedure to avoid mutual collisions betweem MAVs has been experimentally verified during 200 runs of PSO optimization with 40 particles and 70 iterations. The total number of motion planning queries was 560,000. The results shown that 96.8% of all collisions amongst MAVs can be resolved this way. The remaining 3.2% of situations (e.g. if more then one collision is detected in a plan of a pair of MAVs) are again resolved by shifting $p_j(t)$ to such a position $p''_j(t)$ that is at a sufficient distance before the location of the detected colision. Due to the rare occurrence of such a situation, the convergence of the PSO process is decreased only slightly. Nevertheless, any exhaustive multi-robot coordination approach may be utilized in applications where the convergence decrease would be unacceptable, but usually at the cost of growth of computational complexity.

After collision-free 3D paths are computed by VG, the controller described in the Section 6 is used to follow them. If any of the constraints is violated during path following (e.g. localization constraints are violated), the paths are shortened to such a position $p''_j(t)$, where all constraints are still satisfied. Velocity of the PSO particles is then computed as

$$u_j(t) = p''_j(t) - p_j(t-1), \tag{6}$$

where $p''_j(t)$ is the updated position vector and $p_j(t-1)$ is the previous position of the swarm. The

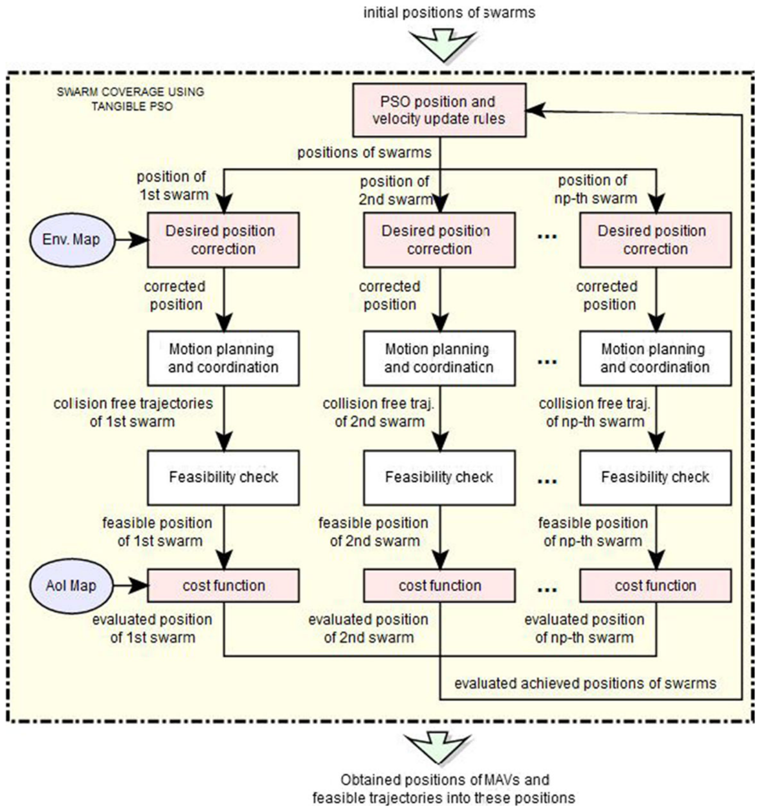


Fig. 6 Scheme of the planning system for the environment coverage by MAV swarms stabilized with the visual relative localization

velocity $u_j(t)$ is then used in the next iteration of the Tanglible PSO to compute new positions of the swarms.

Besides the localization constraints, the motion constraints and the collisions are tested during the simulation. Although VG provides collision-free solutions, it could be difficult to follow them by MAVs due to motion constraints. In the above mentioned statistics of 200 runs of the PSO algorithm with 40 particles and 70 iterations, a collision with obstacles due to the deviation from the followed path was detected in 0.06% of runs of the motion planning algorithm. The prediction of an impending collision again resulted in the change of the components of the PSO vector in Eq. 6.

Another approach to verify reachability between swarm positions is to employ a full motion planning technique where motion model of MAVs and swarm constraints are integrated. For example, sampling-based methods like Rapidly Exploring Random Trees [38] or the method in [39] seems to be appealing. Nevertheless, even the utilization of simple visibility graph technique enables us to show usefulness of motion planning integrated within the evolutionary optimization. Such an algorithm enables to steer MAVs directly using results of the optimization. This can be understood as a novel approach to multi-objective optimization, where a motion planning technique is integrated directly into the core of the optimization engine.

We should again emphasize that the proposed method is not aimed at finding both the optimal swarm distribution and the optimal trajectories from the initial depot. Instead, only the swarm distribution is optimized according to the cost function in Eq. 1 and Tangible PSO ensures that the provided trajectories are feasible. This is motivated by the fact that the motion of MAV swarms in real applications is strongly influenced by the necessity of maintaining direct visibility required by the relative localization system and this paper is focused on this aspect. Later smoothing and post-processing process, which would lead to a sub-optimal solution of the swarm deployment task is possible, but it exceeds the content of this paper.

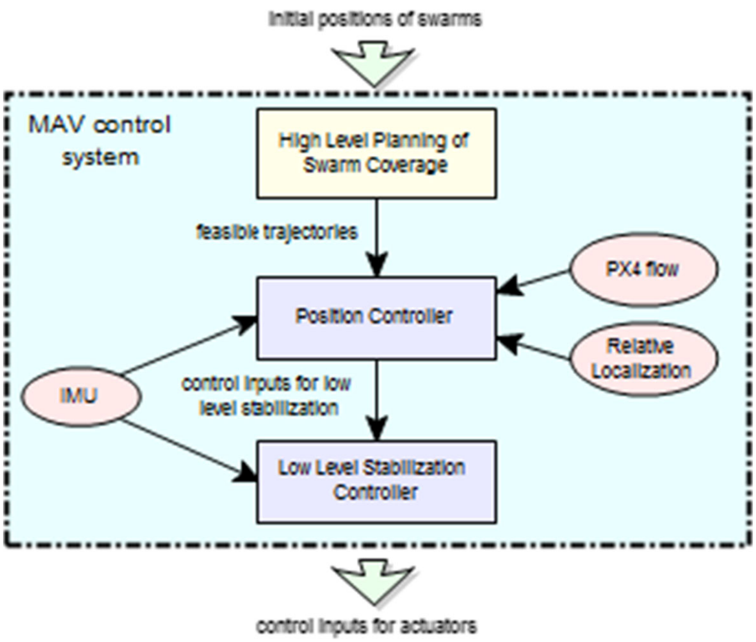


Fig. 7 Scheme of MAV control system

6 Control and Stabilization of MAVs

The trajectories for all MAVs obtained by the high-level PSO planning of swarm coverage are used as an input for the position controller. In the simplified scheme in Fig. 7, it is supposed that the trajectories are feasible for the employed MAVs, i.e., they are collision-free, and motion and relative localization constraints are satisfied. In case of changing environment with dynamic obstacles, an additional obstacle detection and avoidance mechanism need to be integrated into the position controller, which is beyond range of this paper.

As highlighted in Fig. 7, three different sensors are used for position control in the proposed system. The position error (deviation from the desired position on the trajectory and the desired relative position to neighbors in the swarm) is obtained from the vision relative localization system, which is described in Section 3, and by integration of velocity given by the PX4Flow sensor.[1] The odometry provided by the PX4Flow sensor is sufficiently precise in short-term missions (tens of meters). In larger areas, GPS signal can be fused with the information from the relative localization module. As mentioned, the proposed method cannot rely solely on the GPS due to insufficient precision and reliability for stabilization of compact MAV groups.

The velocity error (deviation from the desired velocity given by the preplanned trajectory or deviation from zero velocity if the swarm is hovering at the static positions defined by the final swarm distribution) is provided by the PX4Flow sensor. PX4Flow sensor measures relative velocity between MAV and ground via an optical flow in images captured by the down-oriented camera. The main advantage of this approach lies in possibility of control of strings of relatively stabilized MAVs without increasing motion oscillations that are often observed in coupled systems. The velocity control loop is faster than the control of desired position, and therefore MAVs relatively stabilized to neighboring MAVs do not follow their motion oscillations. Such approach is efficient in swarm surveillance applications, where fast motion manoeuvres are not required and it is especially appealing for the stabilization in final static positions with desired relative distances between swarm members.

Finally, an inertial measurement unit (IMU) provides information on acceleration and tilt of the MAV body. This information is employed in the designed position controller for the swarm stabilization and also in the low-level attitude stabilization. In the experiments, we rely on the MikroKopter set that includes

[1] http://pixhawk.org/modules/px4flow

a proprietary Flight-CRTL stabilization board with integrated IMU.

In [3], a geometric tracking control approach from [40] was employed in numerical simulations and in experiments with a fleet of MAVs controlled from an external PC. In the outdoor experiments presented in this paper, a different controller is utilized due to limited computational power available onboard of MAVs. This enables distribution of the control and stabilization approach on board of particular vehicles, which is necessary for deployment of large groups of MAVs in outdoor scenarios. The original controller implemented according to [40] is used only in simulations in Figs. 9 and 10, which were originally published in [3].

The new controller, which is described in details in [41], provides a sufficient performance for demonstration the ability of the swarm deployment and for verification that the obtained trajectories are feasible for the visual relative localization that is employed in control feedback.

7 Results

Experimental results presented in this paper are aimed at verification of the system performance with a fleet of MAVs, while additional simulations of system behaviour may be found in [3]. In [3], mainly analyses of system performance with different setting of its parameters is presented. The convergence of the PSO process and number of iterations needed for achievement of feasible solutions with different algorithm settings is shown there. Small number of PSO iterations is important due to limited computational time and also for obtaining simpler results with small number of waypoints in the trajectories. Such trajectories are flown through faster and may be better smoothed in a post-processing. As a result of these analyses, the PSO parameters $c_1 = 2.5$, $c_2 = 2.0$, $u_{max} = inf$, and $w = 1$ have been chosen for using the algorithm with tangible particles. This setting is used also in the experiments presented in this paper.

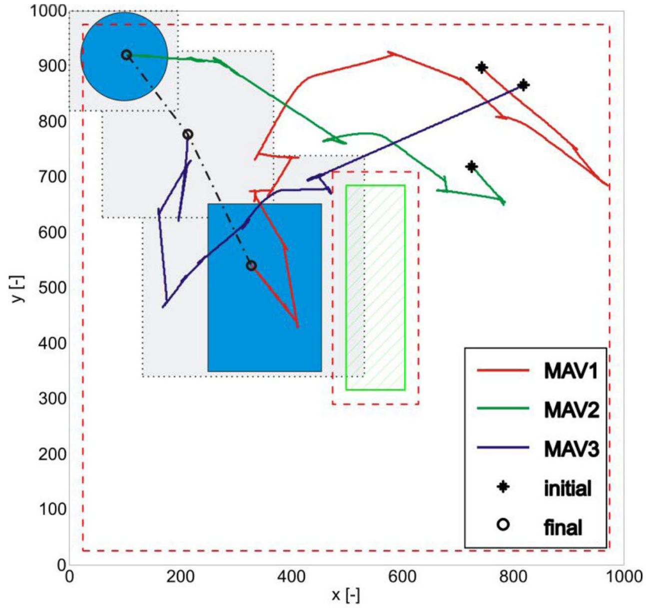
In addition, the influence of constraints given by the visual relative localization and constraints imposed by using tangible particles instead of dimensionless PSO particles, which are used in standard PSO based methods, is discussed in [3]. It is shown in [3] that the non-zero size of particles may be beneficial in the exploitation phase of the PSO process. Contrariwise, the localization constraints help to stabilize the PSO swarm in the initial exploration phase. Finally, it is shown there that the number of particle evaluations, which is number of PSO particles multiply by number of PSO iterations, is not the only parameter that has to be minimized. In our case, the most important aspect is the limited number of PSO iterations, which implies simpler trajectories. Therefore, larger PSO swarms are preferred if using the proposed system (Fig. 8).

7.1 Simulations of the Swarm Deployment

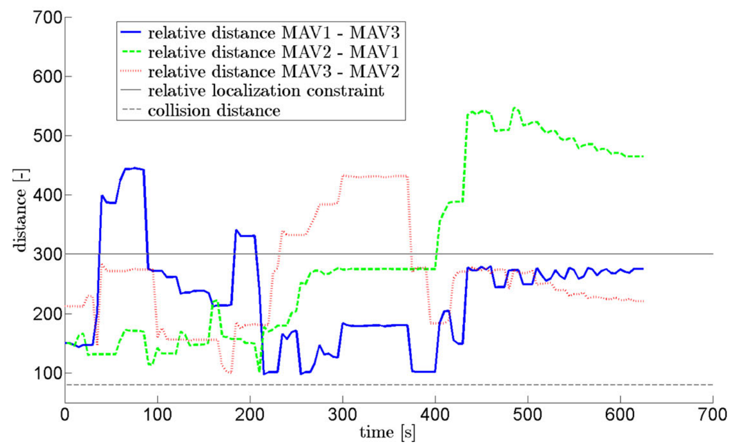
In the first simulation presented in this paper, a swarm of 6 MAVs is deployed in a plane environment without obstacles (see Fig. 9). In the planning of the swarm deployment, it was required that each MAV has to keep localization linkages (constraints of the visual localization have to be satisfied) with at least two neighbors. In Fig. 9, each MAV is connected with other MAVs by red lines if the particular pair is in the relative position, which satisfies the constraint of the visual localization. Four circular areas of interest are denoted in the workspace of robots. At the beginning of the simulation, the MAV swarm is randomly initialized in a circle that represents its starting area. The diameter of the starting area is equal to the maximal localization distance. Since an omnidirectional sensor for relative localization is considered in this simulation, the swarm of MAVs being initialized in such circle always satisfies the localization constraints.

In the experiment, the PSO optimization was run with 20 iterations and 40 PSO particles. The swarm distribution with the lowest cost function was found in the 17-th population. Only the history of movement of the best particle, which is the particle with the same personal best value as is the obtained global best value, is considered as solution of the *swarm distribution* and *swarm deployment* problems. The complete solution was obtained in 32s using Intel Core i7, 2.4 GHz, 4GB RAM in MATLAB. Progress of the evolution of the best particle (the first 17 iterations) is shown using the motion simulation in Fig. 9. During the simulation, the swarm is emergently splitted into two groups to cover both pairs of the locations of interest. The splitting was not preprogrammed, but it autonomously appeared during the optimization. The obtained sub-swarms form triangles, in which the requirement on two neighbors in the localization

Fig. 8 The testing scenario used for the analyses of the system performance and an example of planned trajectories if the relative localization constraint was employed



(a) The obtained trajectories depicted in the workspace with denoted areas of interest and obstacles



(b) Relative distances between the MAVs during the experiment. The relative distance between MAVs shorter than 70 map units (the lower dotted threshold) is considered as infeasible. Always at least one neighbor of each MAV has to be closer than 300 map units (the upper dotted threshold) to satisfy the relative localization constraint.

distance may be satisfied independently to the second group. If the strong connectivity within the group would be required, another constraint, which requires to regularly check compactness of the graph composed from the actual localization linkages, has to be added in to the algorithm.

In the second simulation, 3 MAVs are reaching positions in an environment with a non-fly zone that was added for verification of the obstacle avoidance. The obstacle avoidance algorithm is part of the swarm shape optimization. The experiment is also aimed at showing the importance of the path planning

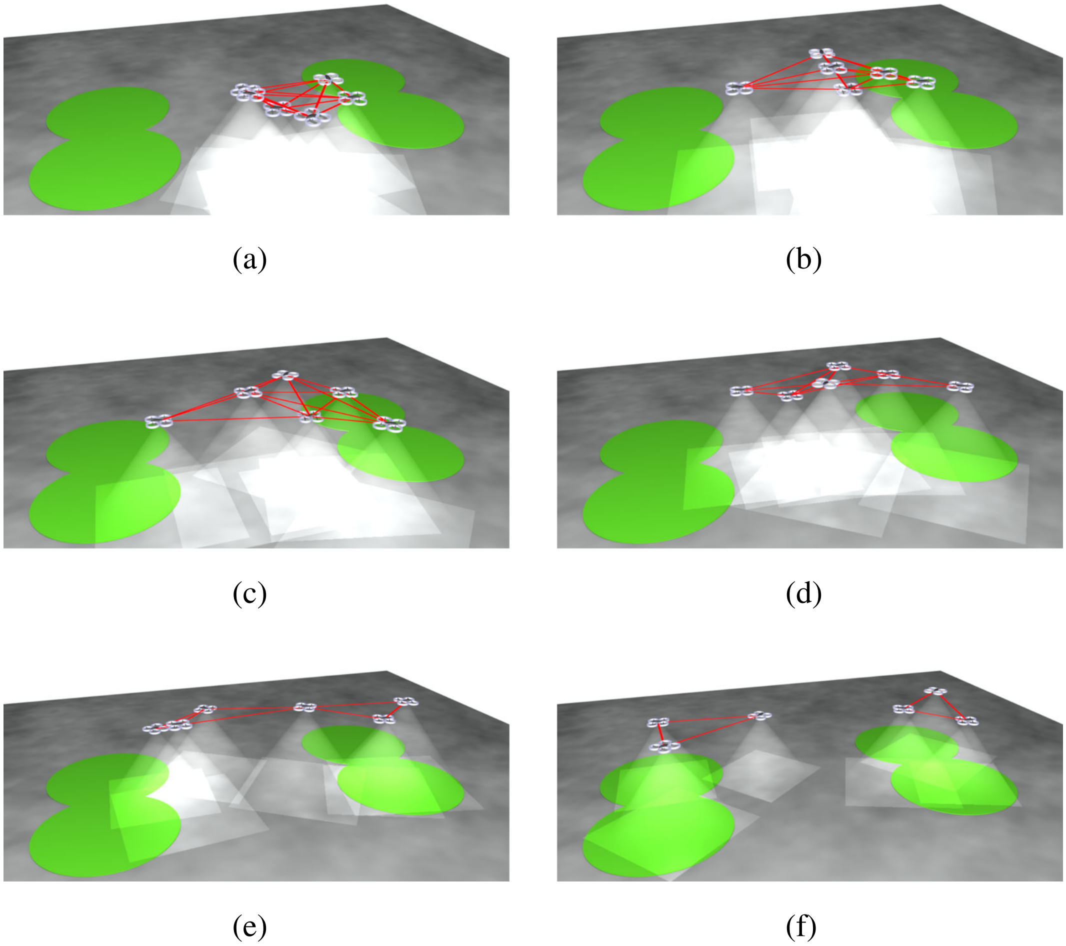


Fig. 9 Simulation with a swarm of 6 MAVs

algorithm, which is included in the core of the optimization. In such environment, the PSO process without the path planning would get stuck because of applied constraints that ensure that MAVs do not enter into the non-fly zone. Without the path planning, all MAVs may end up on the border of the obstacle, while the PSO rules repeatedly force them to continue inside. These deadlocks of the PSO optimization significantly decrease convergence of the algorithm or even freeze its process if the random character of PSO does not enable releasing from these local extrema, as shown in Table 1. The percentage of runs of the algorithm that were finished in a deadlock (the movement of PSO particles was stopped due to the non-fly zone) is stated in the fourth column of the table. The third column of the table denotes the number of PSO runs in which a collision with the non-fly zone was detected in the *Feasibility check* block. The last column shows average cost value of solutions achieved after 70 iterations.

Table 1 Comparison of the performance of the PSO algorithm with and without the motion planning integrated into the PSO optimization

Algorithm specification		Collisions	Deadlocks	Avg. cost
0 neighbor	without MP	98%	31%	2.3×10^5
	with MP	1.3%	0%	1.0×10^5
1 neighbor	without MP	100%	73%	9.2×10^5
	with MP	4.5%	0%	2.4×10^5
2 neighbors	without MP	100%	81%	6.7×10^7
	with MP	9.8%	0%	3.8×10^6

Three scenarios with different required number of neighbors that have to be kept under the relative visual localization are considered

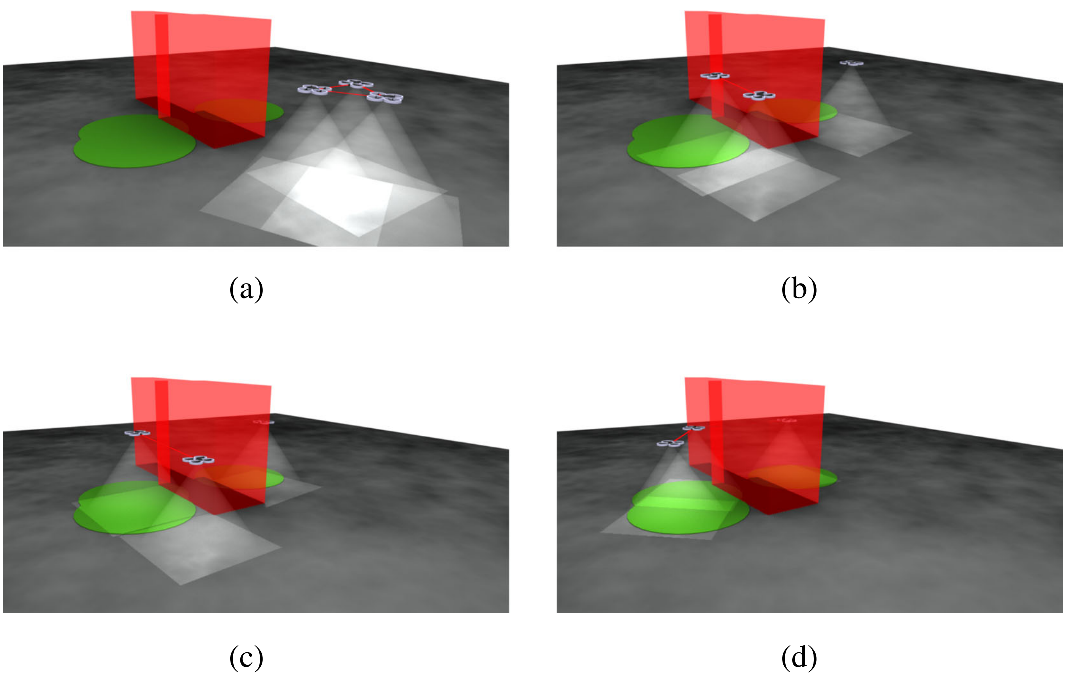
The complete solution was obtained in each run of the algorithm in approximately 28s using the same PC as in the first simulation. The optimization process was slightly faster than in the first simulation although the number of iteration was 3.5 times bigger. This is caused by the smaller swarm used in the second scenario, since the complexity of the feasibility check, which is run in each iteration of the optimization, depends on the number of MAVs.

The results show that even such simple version of the tangible PSO method allows us to obtain feasible solutions of the tackled problem in environment with obstacles and that the algorithm provides sufficient reliability. Moreover, due to the randomized character of PSO and the fact that the optimization problem is solved prior the mission, the PSO process can be run repeatedly from the same initial conditions with results of different quality. The best solution can be then simply chosen from the obtained results comparing their cost.

In experiment in Fig. 11, each MAV has to be connected with at least one neighbor through the localization linkage during their movement into the surveillance location. To compare performance of the swarm distribution algorithm, a standard state-space search was employed for finding a solution with MAVs flying above the areas of interest (see Fig. 10). This simple searching process considers only the quality of the final swarm distribution, i.e., it optimizes only the cost function (1), but the existence of feasible trajectories to reach this position is not checked. In the presented example, the shape of the non-fly zone do not allow to reach this solution by the swarm flying from the initial depot, while keeping the localization constraints. The non-fly zone is too wide for enabling the required relative localization of MAVs following borders of the zone each on the opposite site.

The solution found by the proposed PSO-based method does not cover the areas of interest completely (see Fig. 11), but a feasible solution of the swarm deployment process is guaranteed. In the sequence of snapshots from the simulation, MAVs flying in relative positions feasible for their relative localization are connected by the red lines.

As mentioned in the description of the algorithm, the same kinematic model and MAV controller is used within the swarm planning and in the consequent realization of the plan (either with real robots or in presented simulations). An example of control inputs



(a) (b) (c) (d)

Fig. 10 Simulation of swarm movement into a location found by a simple state space search. The solution does not guarantee that the direct visibility is kept during the movement (it is violated in **b**,**c**,**d**)

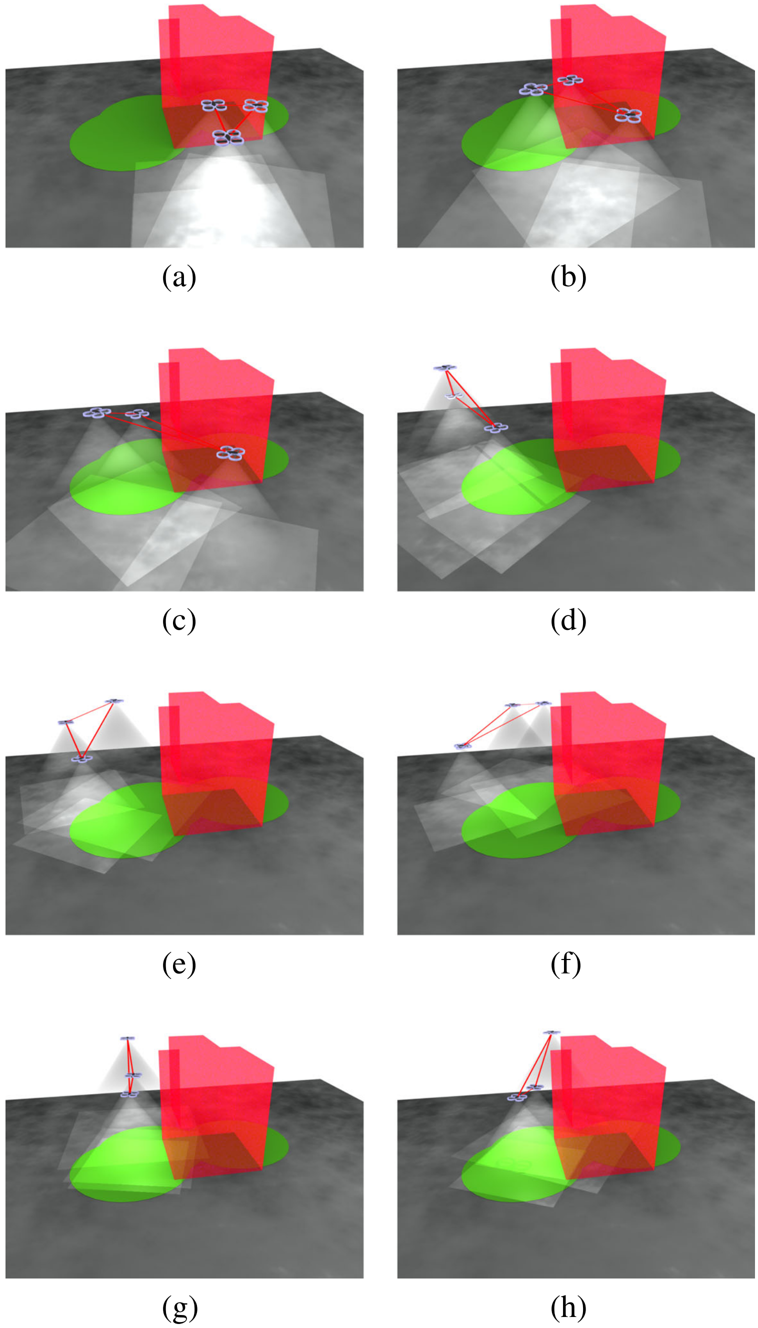


Fig. 11 Simulation with a swarm of 3 MAVs

and state variables of one of the MAVs of the group during its flight between two consequent positions that are generated by the PSO rules is shown in Fig. 12. In the selected trajectory segment, no obstacles or interferences with neighbors occur and the MAV can follow a straight path simply connecting the actual and desired positions. The actual position is $[x = 0, y = 0, z = 0]$ and the desired position is $[2, -10, 1]$. Figures 12 and 13 show x, y, z positions, Euler angles, velocities and control inputs (moments and forces) of the MAV reaching the desired position proposed by the PSO algorithm. The MAV is navigated into the

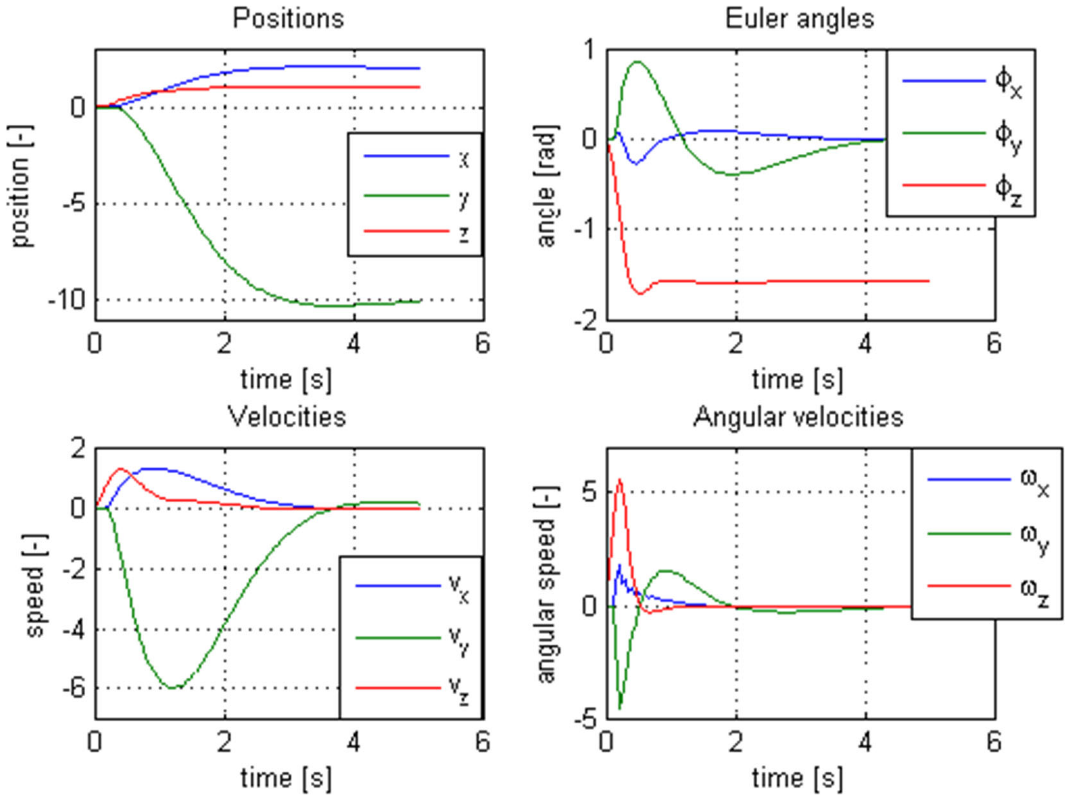
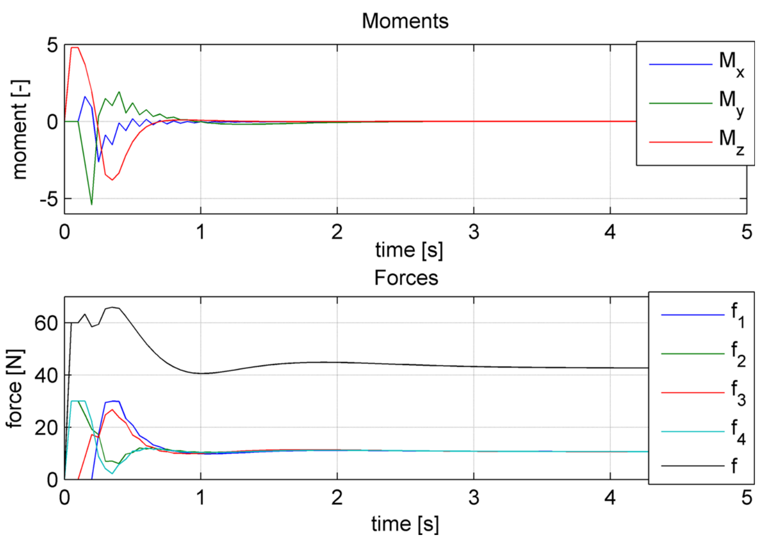


Fig. 12 Progress of positions, velocities, Euler angles and angular velocities during flight between two consequence positions of an MAV. The positions are generated in two consequent steps of the PSO optimization technique

desired location smoothly by the trajectory tracking mechanism, but the overshoots in the position behind the coordinates [2, −10, 1] shown in Fig. 12 have to be considered in the PSO planning. Mainly the obstacle avoidance ability of the system is influenced by this position error.

Fig. 13 Progress of moments and forces used as the inputs of the MAV model in the movement simulation from Fig. 12



7.2 Experimental Evaluation of the Planning Technique in Flight Conditions

The experiment in Fig. 14 was aimed at verification of the ability of the onboard vision system in providing information for the relative localization in flight conditions. It should verify that the linkages between the neighboring MAVs required for the swarm stabilization are kept during execution of the plan obtained by the proposed swarm deployment method. In this experiment, the control feedback of MAVs is realized by the Vicon motion capture system. This system is also important as a ground truth for evaluation of the performance of the relative localization system onboard of MAVs. In these preliminary swarm experiments in laboratory conditions, it was shown that the guess of relative positions of neighbouring vehicles can be continuously provided during the flight.

7.3 Outdoor Experimental Evaluation of the Complete System

The last set of experiments were realized for verification of the designed system in real-world environment

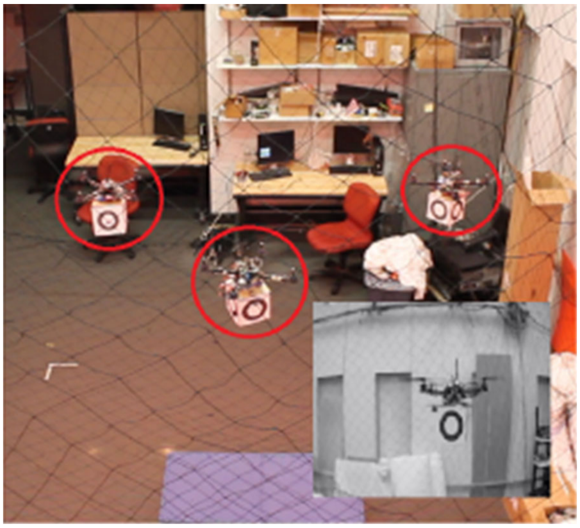


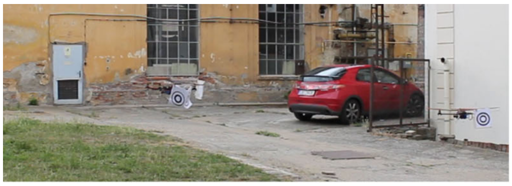
Fig. 14 Experiment with swarm of 3 MAVs following trajectories obtained off-line by the proposed planning algorithm. The *black* and *white* picture is an example of image taken by the onboard localization system. For more details on the experiment, complete multimedia record, and other examples of multi-MAV experiments with visual relative localization conducted at facility of GRASP laboratory, University of Pennsylvania see [3, 42] and [41]

(without the Vicon motion system). In addition, the performance of the MAV stabilization system and its properties in outdoor conditions are shown in these experiments.

Two MAVs that follow an obtained plan of the *swarm distribution* and *swarm deployment* are shown in the first experimental trail (see snapshots from the experiment in Fig. 15 and a video record in [42]). In this experiment, the entire system is employed in the same way as it may be used for the visual surveillance of areas of interest. Prior the mission, a plan of the environment that have to be under surveillance is uploaded into the system with selected areas of interest. These locations may be chosen by security experts. It is not necessary to take into account requirements and limits of the MAV system, since this will be considered during the autonomous planning. Therefore, the day-to-day using of the system do not require assistance of people with robotic expertise.

Once the map with obstacles and areas selected for the surveillance is available, the plan of the *swarm distribution* and *swarm deployment* may be obtained prior the mission. This is advantageous in safety critical applications (surveillance of areas with presence of people, in surrounding of airports, close to state borders, etc.). The plan obtained in advance of the flight may be manually verified in the flight simulator, which is part of the proposed system (the simulation engine is used within the motion planning).

In the next step, the feasible plan is sent to all MAVs and it is used for reaching the locations proper for surveillance (the final *swarm deployment*). As it is shown in Fig. 15, the MAVs are always kept in such positions that allow their relative localization and stabilization. In the experiment, cameras with relatively small viewing angle were used, and therefore the robots must stay in a polyhedron (pyramid) with apex in position of the camera of the neighbour, height 3m, and angle between its faces 20 degrees. In addition, the relative distance between MAVs needs to be bigger than 1.2m. The experiment shows that although these requirements significantly constraint the PSO optimization process, the algorithm provides feasible solution of the task, which may be safely followed by the robots. We should highlight that the obtained trajectories are used without any post-processing. A simple smoothing algorithm would increase quality of the solution of *swarm distribution*, without any



(a) 0s

(b) 13s

(c) 24s

(d) 34s

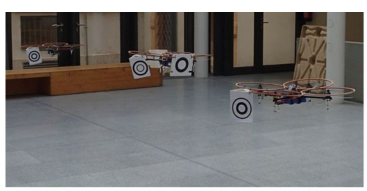
(e) 45s

Fig. 15 Experiment with two relatively stabilized MAVs following the plan obtained by the proposed method

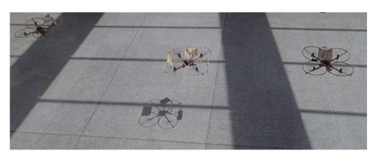
negative effect on the *swarm deployment*, but this is beyond the scope of this paper.

In the second experiment, several examples of different solutions of static *swarm deployment* are tested (see Fig. 16 for pictures from experiments in different environments). The aim of these experimental trails is to show performance of the designed system and the possibility of swarm stabilization. Again, all MAVs

Fig. 16 Examples of different deployments of the swarm, which is stablized under the visual relative localization



(a)



(b)

(c)

are equipped with the same system of visual relative localization. The plan for robots (desired relative positions) is obtained in advance of the mission and it is consequently distributed within the team. The motion and relative localization constraints are satisfied in these desired positions.

8 Conclusion and Future Work

A novel approach for deployment of multi-MAV teams in the application of cooperative surveillance is presented in this paper. The proposed method based on visual onboard relative localization is verified in a feasibility study approved by numerous simulations and hardware experiments demonstrating utilization of the swarm deployment in various scenarios. In addition, the paper provides a study of performance of the evolutionary optimization method that uses tangible particles instead of commonly used dimensionless particles. During the optimization, the movement of particles satisfies motion constraints enforced by deployment of MAVs in the real world missions. The utilization of the optimization method inspired by the swarm intelligence for steering the real 3D swarms of MAVs brings interesting results, which were not observed in simulations of dimensionless swarm particles. Therefore, the proposed approach of swarm stabilization via the local sensory information may act as a test-bed for the study of swarming behaviour and evolution in artificial systems.

As a future work, we plan to combine the PSO optimization with a motion planning technique such as the Rapidly-Exploring Random Tree (RRT), which should significantly improve time required for finding the solution and mainly it should enable to solve more complicated situations in complex maps. In addition, employing a motion planning approach could solve the problem that a proof of convergence (even the probabilistic one) for PSO does not exist.

Acknowledgments The work has been supported by CTU grant no. SGS15/157/OHK3/2T/13 and by the Grant Agency of the Czech Republic under grant no. P103-12/P756. Access to computing and storage facilities owned by parties and projects contributing to the National Grid Infrastructure MetaCentrum, provided under the programme “Projects of Large Infrastructure for Research, Development, and Innovations” (LM2010005), is greatly appreciated.

References

1. Krajnik, T., Nitsche, M., Faigl, J., Vanek, P., Saska, M., Preucil, L., Duckett, T., Mejail, M.: A practical multirobot localization system. In: Accepted by Journal of Intelligent & Robotic Systems (2014)
2. Faigl, J., Krajník, T., Chudoba, J., Preucil, L., Saska, M.: Low-cost embedded system for relative localization in robotic swarms. In: Proc. of IEEE International Conference on Robotics and Automation (2013)
3. Saska, M., Chudoba, J., Precil, L., Thomas, J., Loianno, G., Tresnak, A., Vonasek, V., Kumar, V.: Autonomous deployment of swarms of micro-aerial vehicles in cooperative surveillance. In: 2014 International Conference on Unmanned Aircraft Systems (ICUAS) (2014)
4. Schmickl, T., Crailsheim, K.: Trophallaxis within a robotic swarm: bio-inspired communication among robots in a swarm. Autonom. Robots **25**, 171–188 (2008)
5. Teacy, W., Nie, J., McClean, S., Parr, G.: Maintaining connectivity in uav swarm sensing. In: IEEE GLOBECOM Workshops (2010)
6. Berman, S., Halasz, A., Hsieh, M., Kumar, V.: Optimized stochastic policies for task allocation in swarms of robots. IEEE Trans. Robot. **25**(4), 927–937 (2009)
7. Liu, W., Winfield, A., Sa, J., Chen, J., Dou, L.: Strategies for energy optimisation in a swarm of foraging robots. In: Swarm Robotics, vol. 4433, pp. 14–26 (2007)
8. Hamann, H., Worn, H.: A framework of space-time continuous models for algorithm design in swarm robotics. Swarm Intell. **2**, 209–239 (2008)
9. Winfield, A., Liu, W., Nembrini, J., Martinoli, A.: Modelling a wireless connected swarm of mobile robots. Swarm Intell. **2**, 241–266 (2008)
10. Saska, M., Vakula, J., Preucil, L.: Swarms of micro aerial vehicles stabilized under a visual relative localization. In: ICRA2014: Proceedings of 2014 IEEE International Conference on Robotics and Automation (2014)
11. Kumar, M., Garg, D., Kumar, V.: Segregation of heterogeneous units in a swarm of robotic agents. IEEE Trans. Autom. Control **55**(3), 743–748 (2010)
12. Saska, M.: MAV-swarms: Unmanned aerial vehicles stabilized along a given path using onboard relative localization. In: Proceedings of 2015 International Conference on Unmanned Aircraft Systems (ICUAS) (2015)
13. Turpin, M., Michael, N., Kumar, V.: Trajectory design and control for aggressive formation flight with quadrotors. Autonom. Robots **33**(1–2), 143–156 (2012). [Online]. Available: doi:10.1007/s10514-012-9279-y
14. Bennet, D.J., McInnes, C.R.: Verifiable control of a swarm of unmanned aerial vehicles. J. Aerospace Eng. **223**(7), 939–953 (2009)
15. Barnes, L., Garcia, R., Fields, M., Valavanis, K.: Swarm formation control utilizing ground and aerial unmanned systems. In: IEEE/RSJ International Conference on Intelligent Robots and Systems (2008)
16. Holland, O., Woods, J., Nardi, R., Clark, A.: Beyond swarm intelligence: The UltraSwarm. In: IEEE Swarm Intelligence Symposium, pp. 217–224 (2005)

17. Brkle, A., Leuchter, S.: Development of micro uav swarms. In: Autonome Mobile Systeme 2009, ser. Informatik aktuell, pp. 217–224 (2009)
18. Cai, N., Xi, J.-X., Zhong, Y.-S.: Brief paper swarm stability of high-order linear time-invariant swarm systems. Control Theory Appl. IET **5**(2), 402–408, 20 (2011)
19. Cheah, C.C., Hou, S.P., Slotine, J.J.E.: Region-based shape control for a swarm of robots. Automatica **45**(10), 2406–2411 (2009)
20. Cortes, J., Martinez, S., Karatas, T., Bullo, F.: Coverage control for mobile sensing networks. IEEE Trans. Robot. Autom **20**(2), 243–255 (2004)
21. Renzaglia, A., Doitsidis, L., Martinelli, A., Kosmatopoulos, E.: Adaptive-based distributed cooperative multi-robot coverage. In: American Control Conference (ACC) (2011)
22. Renzaglia, A., Doitsidis, L., Martinelli, A., Kosmatopoulos, E.: Adaptive-based, scalable design for autonomous multi-robot surveillance. In: IEEE CDC (2010)
23. Mathews, E., Graf, T., Kulathunga, K.S.S.B.: Biologically inspired swarm robotic network ensuring coverage and connectivity. In: 2012 IEEE International Conference on Systems, Man, and Cybernetics (SMC), pp. 84–90 (2012)
24. Liu, J.-z., Wang, B.-l., Ao, J.-y., Wang, S., Wu, Q.: An immune-swarm intelligence based algorithm for deterministic coverage problems of wireless sensor networks. J. Central South Univ. **19**(11), 3154–3161 (2012)
25. Maza, I., Ollero, A.: Multiple uav cooperative searching operation using polygon area decomposition and efficient coverage algorithms. In: Alami, R., Chatila, R., Asama, H. (eds.) Distributed Autonomous Robotic Systems 6, pp. 221–230. Springer, Japan (2007)
26. Ha, T.: The UAV Continuous Coverage Problem. Air Force Institute of Technology (2010)
27. Schwager, M., Julian, B.J., Rus, D.: Optimal coverage for multiple hovering robots with downward facing cameras. In: IEEE ICRA (2009)
28. Doitsidis, L., Renzaglia, A., Weiss, S., Kosmatopoulos, E., Scaramuzza, D., Siegwart, R.: 3d surveillance coverage using maps extracted by a monocular slam algorithm. In: IEEE/RSJ IROS (2011)
29. Saska, M., Hess, M., Schilling, K.: Hierarchical spline path planning method for complex environments. In: Proc. of the 4th International Conference on Informatics in Control, Automation and Robotics (2007)
30. Saska, M., Hess, M., Schilling, K.: Voronoi strains - a spline path planning algorithm for complex environments. In: Proc. of the IASTED conference on Artificial Intelligence and Applications (2007)
31. Mouret, J.-B., Doncieux, S.: Encouraging behavioral diversity in evolutionary robotics: An empirical study. Evol. Comput. **20**(1), 91–133 (2012)
32. Vonasek, V., Saska, M., Kosnar, K., Preucil, L.: Global motion planning for modular robots with local motion primitives. In: IEEE ICRA (2013)
33. Saska, M., Vonasek, V., Krajnik, T., Preucil, L.: Coordination and navigation of heterogeneous MAV–UGV formations localized by a ‘hawk-eye’-like Approach Under A Model Predictive Control Scheme. Int. J. Robot. Res. **33**(10), 1393–1412 (2014)
34. Saska, M., Krajnik, T., Vonasek, V., Kasl, Z., Spurny, V., Preucil, L.: Fault-tolerant formation driving mechanism designed for heterogeneous MAVs-UGVs groups. J. Intell. Robot. Syst. **73**(1–4), 603–622 (2014)
35. Saska, M., Krajnik, T., Vonasek, V., Vanek, P., Preucil, L.: Navigation, localization and stabilization of formations of unmanned aerial and ground vehicles. In: Proceedings of 2013 International Conference on Unmanned Aircraft Systems (2013)
36. Kennedy, J., Eberhart, R.: Particle swarm optimization. Proc. Int. Conf. Neural Netw. IEEE **4**, 1942–1948 (1995)
37. O'Rourke, J.: Art Gallery Theorems and Algorithms. Oxford University Press (1987)
38. LaValle, S.M.: Rapidly-exploring random trees: A new tool for path planning. In: TR 98-11, Computer Science Dept. Iowa State University (1998)
39. Turpin, M., Michael, N., Kumar, V.: Concurrent assignment and planning of trajectories for large teams of interchangeable robots. In: International Conference on Robotics and Automation. Karlsruhe, Germany (2013)
40. Lee, T., Leoky, M., McClamroch, N.: Geometric tracking control of a quadrotor uav on se(3). In: 49th IEEE Conference on Decision and Control (CDC) (2010)
41. Saska, M., Baca, T., Thomas, J., Chudoba, J., Preucil, L., Krajnik, T., Faigl, J., Loianno, G., Kumar, V.: System for deployment of groups of micro aerial vehicles in gps-denied environments using onboard visual relative localization. Accepted for Autonomous Robots (2016)
42. Movies, Movies of experiments of the mav cooperative surveillance. http://imr.felk.cvut.cz/mavsurveillance/ [online] [cit. 2014-12-12]

Martin Saska received his MSc. degree at Czech Technical University in Prague, 2005, and his Ph.D. degree at University of Wuerzburg, Germany, within the PhD program of Elite Network of Bavaria, 2009. Since 2009, he is a research fellow at Czech Technical University in Prague, where he is currently head of Multi-robot Systems group. In 2008, he was a visiting scholar at University of Illinois at Urbana-Champaign. He is an author or co-author of more than 40 publications in peer-reviewed journals and conferences. His interests include cooperation and control of multi-robot systems, formation flying, and bio-inspired swarms.

Vojtěch Vonásek obtained his M.Sc. degree in 2008 from Faculty of Electrical Engineering, Czech Technical University in Prague. Currently, he is a Ph.D. student at the Department of Cybernetics at the same faculty. His research interests include motion planning for complex robots, physical simulation in robotics and navigation of mobile robots.

Jan Chudoba received his MSc. degree at Czech Technical University in Prague, 2004. Currently, he is a research fellow at Department of Cybernetics. He has >20 publications in robotics. His interests include navigation of mobile robots, mapping and sensor data fusion.

Justin Thomas received his B.S. in Mechanical Engineering from LeTourneau University in May 2011 and graduated Summa Cum Laude. While at LeTourneau University, Justin researched propulsion systems for fixed-wing aircraft and control systems for biomechanical testing fixtures. In the Fall of 2012, Justin joined the University of Pennsylvania as a Ph.D. student in the department of Mechanical Engineering and Applied Mechanics and as a member of the GRASP Lab under the guidance of Professor Vijay Kumar. At the University of Pennsylvania, Justin is researching aerial manipulation and vision-based control using Micro Aerial Vehicles.

Giuseppe Loianno received B.Sc and M.Sc in Automation Engineering both with honours at University of Naples Federico II in December 2007 and February 2010, respectively. From August 2008 till March 2009 he has been an exchange student at KTH (Royal Insitute of Technology) in Stockholm, then from September 2009 till March 2010 he developed his master thesis at ETH Zurich at the ASL laboratory focusing on Micro Aerial Vehicles under the supervision of Prof. Dr. Davide Scaramuzza. He received his Ph.D in computer and automatic engineering focusing in Robotics in May 2014 in the PRISMA Lab group, led by Prof. Dr. Bruno Siciliano. From April 2013 he worked for 14 months with the Grasp Lab at University of Pennsylvania, supervised by Prof. Dr. Vijay Kumar, where currently he is a postdoctoral researcher. His research interests include visual odometry, sensor fusion and visual servoing for micro aerial vehicles.

Vijay Kumar is the UPS Foundation Professor in the School of Engineering and Applied Science at the University of Pennsylvania. He received his Bachelors of Technology from the Indian Institute of Technology, Kanpur and his Ph.D. from The Ohio State University in 1987. He served as the Deputy Dean for Research in the School of Engineering and Applied Science from 2000-2004. He directed the GRASP Laboratory, a multidisciplinary robotics and perception laboratory, from 1998-2004. He was the Chairman of the Department of Mechanical Engineering and Applied Mechanics from 2005-2008 and the Deputy Dean for Education in the School of Engineering and Applied Science from 2008-2012. He was the assistant director for robotics and cyber physical systems at the White House Office of Science and Technology Policy from 2012-2014. Dr. Kumar's research interests are in robotics, specifically multi-robot systems, and micro aerial vehicles. He has served on the editorial boards of the IEEE Transactions on Robotics and Automation, IEEE Transactions on Automation Science and Engineering, ASME Journal of Mechanical Design, the ASME Journal of Mechanisms and Robotics and the Springer Tract in Advanced Robotics (STAR). He is the recipient of the 1991 National Science Foundation Presidential Young Investigator award, the 1996 Lindback Award for Distinguished Teaching (University of Pennsylvania), the 1997 Freudenstein Award for significant accomplishments in mechanisms and robotics, the 2012 ASME Mechanisms and Robotics Award, the 2012 IEEE Robotics and Automation Society Distinguished Service Award and a 2012 World Technology Network Award. He has won best paper awards at DARS 2002, ICRA 2004, ICRA 2011, RSS 2011, and RSS 2013, and has advised doctoral students who have won Best Student Paper Awards at ICRA 2008, RSS 2009, and DARS 2010.