Complex manoeuvres of heterogeneous MAV-UGV formations using a model predictive control

Vojtech Spurny  
Faculty of Electrical Engineering  
Czech Technical University in Prague  
Email: vojtech.spurny@fel.cvut.cz

Tomas Baca  
Faculty of Electrical Engineering  
Czech Technical University in Prague  
Email: tomas.baca@fel.cvut.cz

Martin Saska  
Faculty of Electrical Engineering  
Czech Technical University in Prague  
Email: martin.saska@fel.cvut.cz

Abstract—A problem of motion planning and coordination of compact formations of ground and aerial robots will be tackled in this paper. The scenarios when the formation composed from Unmanned Ground Vehicles (UGVs) and Unmanned Aerial Vehicles (UAVs), in particular Micro Aerial Vehicles (MAVs), has to reverse the direction of movement to fulfill task of collision-free motion to a target zone will be solved. The presented motion planning and stabilization approach provides an effective technique to enable deployment of closely cooperating teams of robots in outdoor as well as indoor environment. The formation to target region problem is solved using a Model Predictive Control (MPC) methodology and the formation driving concept is based on a virtual-leader-follower approach. The mentioned MPC based process is used for trajectory planning and control of a virtual leader and also for control and stabilization of followers (MAVs and UGVs). The proposed approach is verified with numerous simulations and hardware experiments.

MULTIMEDIA MATERIAL
A video attachment to this work is available at:  
http://mrs.felk.cvut.cz/mmar2016plan

I. INTRODUCTION
With advancing technological progress, mobile robots become more spread than ever and also multi-robot systems become frequently used because of their various potential usages in both civilian and military applications such as search and rescue missions [1], forest fire detection [2], surveillance [3]. One of the currently investigated problems dealing with multi-robot systems is coordination of formations of robots and their motion planning into a target zone. In this task, a group of robots has to find and follow a collision free trajectory while it maintains a desired shape of the formation.

Various formation control strategies were studied for solving this problem and can be divided into three main groups: leader-follower [4], virtual structure [5], and behaviour approach [6]. In the leader-follower structure technique, one or more robots in the formation are assigned as leaders while the rest of the formation members as followers. However, this concept is not robust, if the leading robot fails the formation fails too. The dependence of the system on the single point failure can be solved by applying a virtual leader strategy. This strategy increases the robustness of the system, since in case of a failure of any robot, the remaining members of the team can continue with their task without the need to select an alternative leader. In the virtual structure approach, the entire formation is considered as a single virtual rigid body with fixed geometric relationship among robots. In the behaviour-based approach, local desired behaviours are assigned to each robot (e.g. collision avoidance, formation keeping). The final behaviour of each robot is computed as a combination of desired behaviours. The main advantage of this structure is in decentralisation of the problem of the controlling multi-robot system, while the disadvantages is that it is hard to proof stability of the formation.

The formation movement is limited by motion capabilities of robots in the formation, such as control input constraints. The optimization-based methods are often used for solving formation control problems with these constrains. One of the most popular optimization-based technique is Model Predictive Control (MPC), which is also called as receding horizon control [7], [8]. The main idea of the MPC is to find the optimal future control actions according to a prediction of the system behaviour over receding horizon online every time a state measurement or estimate became available. A more detailed description of MPC methods can be found in [9].

Here, we focused on applications, in which localization of members of the formation based on available Global Navigation Satellite System (GNSS) is insufficient due to relative distances of team members that are smaller than precision of GNSS or due to absence of GNSS signal. To solve the formation flying without GNSS, we propose to rely on an onboard vision system (small light-weight cameras) employed for relative localization of neighbouring members of the formation (see [10], [11] for details on this system). In this approach, it is crucial to keep direct visibility between
the members of the formation during their movement. Any breakage of such visibility would lead to breaking coherence of the formation, since the information on relative positions of neighbours obtained by the visual-based localization process is directly used in the control feedback of the formation stabilization algorithm.

II. PROBLEM FORMULATION

This paper relies on the approach that we have explained in [12]. This technique solves heterogeneous UGV-MAV formation to target region problem using the MPC method based on the virtual leader-follower approach as the formation control strategy. Further, the relative localization of neighbouring members of the formation is obtained by the mentioned onboard vision system. The approach in [12] does not allow reverse of direction of the formation driving if it is required by an obstacle in the environment. Using the method in [12], all followers behind the leader continue with the direction of their movement to the state where the virtual leader changed the direction of the movement, this may lead into a collision with robots that are in different distances to the position of the virtual leader as shown in Fig. 2. To solve this problem, it is necessary to change movement direction of all followers in the same moment.

The reverse of direction of the formation driving is necessary in situations, where a complex manoeuvring of the formation in a cluttered environment is required, such as e.g. turning 180 degrees at the end of a narrow row. The algorithm presented in this paper enables complex manoeuvres including multiple reverse driving for accomplishing the task. For the complex manoeuvring, we propose to employ an additional virtual leader, which is assigned for leading the formation in parts of the trajectory, where the motion direction of the formation is reversed. This leader is located in rear of the formation and on its axis. Leading role of the virtual leaders is always switched when the polarity of the velocity of the formation is changed (see Fig. 3).

The idea of the multiple leaders switching is not new. For instance, the leader switching for tele-operation of Unmanned Aerial Vehicles (UAVs) is studied in [13], where the pilot must control the team of UAVs while maintaining control of his own plane. In [14], a planning approach is presented that automatically determines when to switch leadership to different robot during movement of the formation into the goal location, while minimizing a given cost function that penalizes leader switching and deviations from the desired shape of the formation. In our paper, the motivation for using the multiple virtual leaders is different. Our approach enables to solve complex manoeuvres including multiple reverse driving for accomplishing multi-robot tasks in cluttered GPS-denied environment that cannot be solved by state-of-the-art methods.

III. PRELIMINARIES

Let us firstly define some preliminaries necessary for description of the approach solving the task in which a compact formation of MAVs and UGVs has to reach a target region.

Let denote configuration of the virtual leaders $L_1$, $L_2$, and $n_r$ numbers of the followers at time $t$ as $\psi_j(t) = [x_j(t), y_j(t), z_j(t), \varphi_j(t)]^T$, where $j \in \{L_1, L_2, 1, \ldots, n_r\}$. The configuration of the $j$-th robot is composed by its position in Cartesian coordinates $p_j(t) = [x_j(t), y_j(t), z_j(t)]^T$ and by its heading $\varphi_j(t)$.

The kinematic model for any robot $j$ is described by $\dot{x}_j(t) = v_j(t) \cos(\varphi_j(t))$, $\dot{y}_j(t) = v_j(t) \sin(\varphi_j(t))$, $\dot{z}_j(t) = w_j(t)$, and $\dot{\varphi}_j(t) = K_j(t)v_j(t)$, where $v_j(t)$ and $w_j(t)$ represent forward velocity and ascent velocity, respectively.

\textsuperscript{1}Symbol $L$ will be used in description of the method if it is not necessary to distinguish between the particular virtual leaders or for description of trajectories for both leaders composed together.
neighbors Map, K and Fig. 5. Illustration of the motion planning and stabilization approach.

| vehicle mechanical capabilities as is limited to zero. Each robot movement is limited by the z control inputs coordinates MAVs in 3D should be described using three parameters yaw, these curvilinear coordinates to Cartesian coordinates at time where ψ

Formation Driving

In the proposed approach, the relative state of (ψ

ψ

formation. All these limitations are considered in trajectory planning of the virtual leader (see [15]).

ψ

virtual leader trajectory in state

j

ψ

virtual leader do not depend only on mechanical capabilities from [15] for examples. For reconstruction the entire trajectory from the vector ΩL its evaluation during the optimization and final realization, the parts of the plan that have the positive polarity of the value of \( v_{L,k} \) are assigned to the first virtual leader and the rest of the plan with the negative polarity to the second virtual leader. The appendixes are considered in the resulting trajectory when the polarity of the value of \( v_{L,k} \) is changed (the leadership is shifted from one virtual leader to the other one) for a smooth continuation of the movement of the formation while keeping the desired shape. Everytime when the leadership is shifted, the relative positions of the followers are recomputed to the new relative position of the leading virtual leader.

To fully describe formation motion, and so the trajectory, in the appendixes, also the velocity of the virtual leaders, denoted

IV. Trajectory planning and stabilization using the MPC

The proposed formation driving system is divided into two blocks (see Fig. 5). In the Virtual leader part, the Trajectory Planning block provides control inputs for the virtual leader and a complete trajectory to the target zone. For this task, the standard model of the predictive control was extended by an additional horizon in which the sampling time \( \Delta t \) is variable between \( M \) transition points. The entire horizon is therefore formed from two horizons. The first horizon \( [t_0, t_0 + N \Delta t] \) is denoted as the control horizon and the second horizon \( (t_0 + N \Delta t, t_0 + (N + M) \Delta t) \) as the planning horizon. The control horizon with the constant sampling time \( \Delta t \) is used to obtain immediate control of the virtual leader. In the planning horizon, lengths of time intervals between \( M \) transition points are also variables taking part in the planning problem to the target zone. The resulting trajectory is used as an input for the second main block, which transforms the plan of the virtual leaders to the desired trajectory for the followers (using Eq. (1)), and for re-initialization of the optimization in the next planning step.

In the Follower block, the Trajectory Following module is responsible for computing a trajectory (control inputs) that avoids collisions with the obstacles and the other members in the formation and that is as close as possible to the desired trajectory provided by the virtual leader. The first \( n \) of the computed control inputs are applied to steer the system.

1) Trajectory planning for the virtual leaders: The trajectory planning problem for both virtual leaders is defined as optimization problem over two time horizons (control horizon and planning horizon). The plan is described as the one vector of decision variables \( \Omega_L = [q_{L,1}, \ldots, q_{L,N}, q_{L,N+1}, \ldots, q_{L,N+M}] \), with \( N+M \) elements, where \( N \) is number of elements in the control horizon and \( M \) in the planning horizon. The elements are defined as \( q_{L,k} = [v_{L,k}, w_{L,k}, K_{L,k}, \Delta t_{L,k}, \omega_{L,k}, K_{L,k}^{App}, K_{L,k}^{App}] \), with \( k \in \{1, \ldots, N+M\} \), where \( v_{L,k}, w_{L,k}, K_{L,k} \) are control inputs, \( \Delta t_{L,k} \) represents duration of these inputs, and \( \omega_{L,k}, K_{L,k}^{App}, K_{L,k}^{App} \) are parameters that represent additional path (appendix) required for smooth changing of the velocity direction

In case of turning of the group, each robot has to move with different value of curvature and velocity for keeping the desired shape of the formation. Therefore, motion constraints of the virtual leader do not depend only on mechanical capabilities of the followers but also on their relative positions in the formation. All these limitations are considered in trajectory planning of the virtual leader (see [15]).

\[
\psi_j(t) = \psi_L(t_p_j) + q_j \begin{bmatrix} \sin(\varphi_L(t_{p_j})) \\ \cos(\varphi_L(t_{p_j})) \\ 0 \\ 0 \\ 0 \end{bmatrix} + h_j \begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \end{bmatrix}, \tag{1}
\]

where \( \psi_L(t_{p_j}) = [x_L(t_{p_j}), y_L(t_{p_j}), z_L(t_{p_j}), \varphi_L(t_{p_j})]^T \) is a configuration of the virtual leader at time \( t_{p_j} \), which represents state of the virtual leader in the past when it was in distance \( p_j \) along the leader trajectory from its actual state \( \psi_L(t) \). Follower state \( \psi_j(t) \) then lies on a line, which is perpendicular to the virtual leader trajectory in state \( \psi_L(t_{p_j}) \), in distance \( q_j \) from this state \( \psi_L(t_{p_j}) \). Further, \( z \) coordinate is set to \( h_j \).

The transition points separate intervals of the trajectory in which the control inputs are constant. This means that in between the transition points the control inputs are constant and in the transition point can be changed.

\[ \Delta t_{L,k} = \Delta t, k \in \{1, \ldots, N\}, \text{since the sampling time is constant in the control horizon}. \]
as \( v_{L,k}^{\text{App}} \), and the motion time, denoted as \( \Delta t_{L,k}^{\text{App}} \), need to be specified. The value of \( v_{L,k}^{\text{App}} \) is used as \( v_{L,k}^{\text{App}} := v_{\max,L} \) for the forward movement and \( v_{L,k}^{\text{App}} := v_{\min,L} \) for the backward movement. The value of \( \Delta t_{L,k}^{\text{App}} \) can be then obtained as \( \Delta t_{L,k}^{\text{App}} := \frac{l_a}{v_{L,k}^{\text{App}}} \), where \( l_a := \max_{j=1,...,n_r} (p_j) \).

The cost function for evaluation of the vector \( \Omega_{L} \), which represents the trajectory of both virtual leaders, is given by

\[
\lambda_{L}(\Omega_{L}) = \alpha \sum_{k=0}^{N+M} \Delta t_{L,k} + \beta \sum_{k=1}^{N+M-1} (v_{L,k} - v_{L,k+1})^2 \\
+ \gamma \sum_{k=1}^{N+M-1} (w_{L,k} - w_{L,k+1})^2 \\
+ \zeta \sum_{k=1}^{N+M-1} (K_{L,k} - K_{L,k+1})^2 \\
+ \eta \sum_{k=1}^{N+M-1} \left( (K_{L,k} - K_{L,k})^2 + (K_{L,k+1} - K_{L,k})^2 \right) \\
+ \kappa \sum_{k=1}^{N+M-1} \left( (w_{L,k} - w_{L,k})^2 + (w_{L,k+1} - w_{L,k})^2 \right) \\
+ \chi \left( \min_{j} \left( \frac{\text{dist}(\Omega_{j}, \Omega_{obs}) - r_{d,L}}{\text{dist}(\Omega_{j}, \Omega_{obs}) - r_{a,L}} \right) \right)^2
\]

The first part is used to minimized the time required to reach the target area. Only the duration of the planning horizon is considered, since the time of control inputs in the control horizon is constant. The next five parts of the cost function are included to make the plan more smooth by minimizing the difference of control inputs between neighbours parts of the trajectory. The last part of \( \lambda_{L}(\cdot) \) represents influence of obstacles close to the planned trajectory. Function \( \text{dist}(\Omega_{L}, \Omega_{obs}) \) provides minimal Euclidean distance from all detected obstacles \( \Omega_{obs} \) to the trajectory. The obstacles that are farther than \( r_{d,L} \) from the planned trajectory do not influence the cost function. By setting parameters \( \alpha, \beta, \gamma, \zeta, \eta, \kappa, \) and \( \chi \), one can be select which type of the trajectory is preferred. For example by increasing parameter \( \chi \), an obstacles near to the trajectory will have higher influence to the cost function and therefore the trajectory further from the obstacles is found. We set all these parameters to 1, in the presented experiments to provide a general behaviour of the method.

The trajectory planning and obstacle avoidance problem is then solved as minimization of the cost function \( \min_{\lambda_{L}(\Omega_{L})} \) subject to sets of inequality constraints. Inequality constraints \( v_{\min,L} \leq v_{L,k} \leq v_{\max,L}, \left| K_{L,k} \right| \leq K_{\max,L}, v_{\min,L} \leq w_{L,k} \leq w_{\max,L}, \left| K_{L,k} \right| \leq K_{\max,L}, v_{\min,L} \leq w_{L,k} \leq w_{\max,L}, \forall k \in \{1,...,N+M\} \), and \( \Delta t_{L,k} \geq 0, \forall k \in \{N+1,...,N+M\} \) limit control inputs. Inequality constraint \( r_{a,L} < \text{dist}(\Omega_{L}, \Omega_{obs}) \) characterizes safety regions around the trajectory and guarantees that the trajectory does not lead to collision with an actually known obstacle. The collisions in appendixes are checked only if the velocity in the precedent and consequent segments of the trajectory differs and so the leadership is switched. Finally, the stability constraint is satisfied if the final state of the trajectory is inside the desired target region.

2) Trajectory following for followers: The trajectory following problem for \( j \)-th member of the formation, where \( j \in \{1,...,n_r\} \), is formulated also as an optimization problem. The trajectory is described by an vector of the decision variables \( \Omega_{j} = [u_{j,1},...,u_{j,N}] \), where \( N \) is length of the control horizon, and evaluated by the cost function

\[
\lambda_{j}(\Omega_{j}) = \alpha \sum_{k=1}^{N} [p_{j,k} - p_{\text{Des}}]^{T} [p_{j,k} - p_{\text{Des}}] \\
+ \beta \sum_{k=1}^{N} (w_{j,k} - w_{j,k})^2 \\
+ \gamma \sum_{k=1}^{N} (w_{j,k} - w_{j,k})^2 \\
+ \zeta \sum_{k=1}^{N} (K_{j,k} - K_{j,k})^2 \\
+ \eta \sum_{k=1}^{N} \left[ (K_{j,k} - K_{j,k})^2 + (K_{j,k+1} - K_{j,k})^2 \right] \\
+ \kappa \sum_{k=1}^{N} \left[ (w_{j,k} - w_{j,k})^2 + (w_{j,k+1} - w_{j,k})^2 \right] \\
+ \chi \left( \min_{\forall k \in \{1,...,n_r\} \setminus \{j\}} \left( \frac{\text{dist}(\Omega_{j}, \Omega_{obs}) - r_{d,j}}{\text{dist}(\Omega_{j}, \Omega_{obs}) - r_{a,j}} \right) \right)^2
\]

The first part of the cost function penalises deviation from the desired positions \( p_{\text{Des}} \), which are computed using Eq. (1). The next three parts of the formula are included to make the plan more smooth by minimizing the difference of control inputs between neighbours parts of the trajectory. The last two parts represent influence of obstacles and other members of the formation close to the planned trajectory in similar way as in Eq. (2). By setting parameters \( \alpha, \beta, \gamma, \zeta, \kappa, \) and \( \chi \), one can again select which type of the trajectory is preferred. We set all these parameters to 1.

Trajectory planning is again solved as minimization of the cost function \( \min_{\lambda_{j}(\Omega_{j})} \) subject to sets of inequality constraints. Inequality constraints \( v_{\min,j} \leq v_{j,k} \leq v_{\max,j}, [K_{j,k}] \leq K_{\max,j}, v_{\min,j} \leq w_{j,k} \leq w_{\max,j}, \forall k \in \{1,...,N\} \), limit control inputs. Inequality constraint \( r_{a,j} < \text{dist}(\Omega_{j}, \Omega_{obs}) \) characterizes safety regions around the trajectory and it guarantees that the trajectory does not lead to collision with any known obstacle. Inequality constraints \( r_{a,j} < \text{dist}(\Omega_{j}, \Omega_{k}) \), where \( k = \{1,...,n_r\} \setminus \{j\} \), guarantee that the trajectory does not lead to collision with other members of the formation.

V. NUMERICAL AND EXPERIMENTAL RESULTS

The Sequential Quadratic Programming (SQP)[16] method is employed for solving the optimization problem in the proposed trajectory planning and stabilization methods. In the experiments, the map of the environment, position of the target region and desired shape of the formation are always known at the beginning of missions. The setting of parameters of the algorithm was \( \Delta t = 0.5s, n = 2, N = 5 \), and \( M = 7 \). For the experimental verification of the proposed method, a set of formation driving scenarios that cannot be solved by the approach in [12] was selected.

In the first simulation presented in Fig. 6, the formation of nine robots (see Table I for the parameters of the formation) has to turn at the end of a road by 180 degrees. The formation cannot simply turn on the spot but a more complicated
(a) Complete plan for both virtual leaders found in the first planning step of the MPC method.

(b) The second virtual leader overtakes the leadership.

(c) The leadership is returned to the first virtual leader.

(d) The paths passed by the members of the formation.

Fig. 6. Snapshots from the experiment where the formation of nine followers has to turn 180 degrees.

TABLE I. CURVILINEAR COORDINATES OF THE FOLLOWERS IN THE FORMATION USED IN THE EXPERIMENT PRESENTED IN Fig. 6.

<table>
<thead>
<tr>
<th>i</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
</tr>
</thead>
<tbody>
<tr>
<td>(p_i , [m])</td>
<td>0</td>
<td>0.6</td>
<td>-0.6</td>
<td>1</td>
<td>0.6</td>
<td>1</td>
<td>1.1</td>
<td>1</td>
<td>1.1</td>
</tr>
<tr>
<td>(q_i , [m])</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>(h_i , [m])</td>
<td>0</td>
<td>0</td>
<td>0.6</td>
<td>1</td>
<td>-0.6</td>
<td>1</td>
<td>-1.2</td>
<td>2</td>
<td>-1.2</td>
</tr>
</tbody>
</table>

The second simulation presented in Fig. 10 shows ability of the proposed approach to find and use a feasible plan in dynamic environment with both static and dynamic obstacles (these obstacles are detected during the movement of the formation). In this simulation, the formation composed of five members has to move to the target region through a narrow corridor. The complete plan for both virtual leaders found in the first planning step is shown in Fig. 10(a) and contains two switches of the leadership. The reaction of the approach to newly detected obstacles is visualized in Fig. 10(d)-10(e).

Finally, in the hardware experiment in Fig. 8, a formation of the ground robot and two MAVs has to move from its initial location into a desired target region using the trajectory planning and formation stabilization approach presented in this paper. The aim of the experiment is to verify the possibility of formation stabilization using only onboard sensors (cameras) and onboard identification patterns. The trajectories obtained in a simulation (called reference trajectory in the picture) and the positions of robots in the hardware experiment obtained by fusion from the onboard position sensors, odometry of the ground robot, and the onboard vision relative localiza-

Fig. 7. The average values of the heading of the followers in rows during the simulation presented in Fig. 6.

Fig. 8. Photos of the platforms used for the hardware experiment.

The work has been supported by CTU grant no. SGS15/157/OHK3/2T/13, MŠMT under grant no. 7AMB16FR017, and by Cesnet under grant no. 569/2015.
Fig. 9. Trajectories of robots in the hardware experiment.

Fig. 10. Snapshots from the experiment in a narrow corridor with obstacles.

(a) The initial position of the formation with a complete plan for both virtual leaders found in the first planning step. The initial plan is denoted by gray points.

(b) The second virtual leader overtakes the leadership.

(c) The leadership is returned to the first virtual leader.

(d) The result of replanning after detection of a static obstacle. The plan found in the previous planning step, when the obstacle was not known, is denoted by gray points.

(e) Applied trajectories after detecting obstacles during the movement to the target region. The motion model of the obstacle was known and this information was used for prediction of the positions of the obstacle in the future. Details on the motion prediction of dynamic obstacles can be found in our previous work in [17].

Fig. 10. Snapshots from the experiment in a narrow corridor with obstacles.

REFERENCES
