Swarming Fixed-Wing Trajectory Planning Using Distributed Model Predictive Control

The format of this research paper is different from standard template after consultation with my supervisor.

Oliver Wykes
Department of Mechanical and Aerospace Engineering
Monash University
Melbourne, Australia
ohwyk1@student.monash.edu

***Abstract*—A distributed model predictive control (MPC) scheme is proposed as a trajectory planner for swarming fixed-wing aircraft. Collision avoidance is implemented using both state constraints and objective function methods, with state constraints found to be problematic when a nonlinear F-16 model is simulated. The use of objective function terms is found to produce desirable obstacle and aircraft avoidance behaviour. Simulation results from the swarming of 32 F-16s in the presence of obstacles demonstrates the potential of this scheme for real-world use.**

I. Introduction

Animals that exhibit swarming behaviour, such as schooling fish and murmurations of starlings, enjoy benefits of their cooperation that could not be achieved alone. This synergy has been identified as a promising new space for robotics, as cooperation between agents can allow them to be more effective and robust to individual failures. Of particular interest is the swarming behaviour of unmanned fixed-wing aircraft, whose high cruise speeds and range give them significant advantages for many of the promising applications of multi-agent systems. These applications include aerial surveillance [1], forest fire monitoring [2], search and rescue [3], nuclear radiation mapping [4], formation of wireless communication networks [5], oil spill tracking [6], and deep-space formation flying [7].

Whereas similar work has been performed for helicopters and multirotor aircraft [8][9], this project focuses on the generation of realistic 3D trajectories for fixed-wing aircraft using distributed MPC. It aims to demonstrate a feasible scheme for real-world use, and highlights shortcomings of state constraint based obstacle avoidance. Fig. 1 illustrates the system hierarchy implemented in this paper, and includes a decision-making layer identified as necessary for future work.

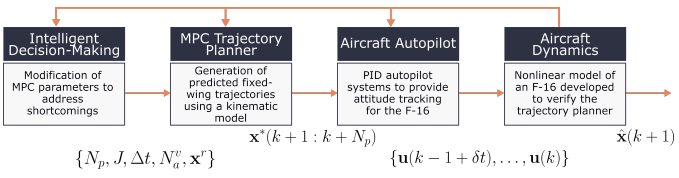


Fig. 1. System hierarchy including a decision-making layer proposed for future work

II. Model Predictive Control

The primary focus of this paper - trajectory planning for swarming fixed-wing aircraft - requires knowledge of model predictive control (MPC). MPC uses a known model of system dynamics, $\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \mathbf{u})$, to predict future motion based on an initial condition $\mathbf{x}^0$ and sequence of control inputs $\{\mathbf{u}(k), \mathbf{u}(k+1), \ldots, \mathbf{u}(k+N_p)\}$. This sequence is found by minimising an objective function $J(\cdot)$ over a finite prediction horizon N_p, and as such MPC is an optimisation-based control scheme. A general form of the MPC objective function is

$$J(\cdot) = S(\mathbf{x}(N_p|k)) + \sum_{l=0}^{N_p} L\left(\mathbf{x}(k+l|k) + \mathbf{u}(k+l|k)\right), \quad (1)$$

where $L(\cdot)$ is the stage cost and $S(\cdot)$ is the terminal cost. The notation $\mathbf{x}(k+l|k)$ represents the predicted state $\mathbf{x}$ at a future time $k+l$ based on an initial condition at time k. From this, the optimal control problem

$$\{\mathbf{x}^*(k+1:k+N_p), \mathbf{u}^*(k:k+N_p-1)\} = \arg\min J(\cdot) \quad (2)$$

can be recursively solved online, with initialisation at the system's current position, $\mathbf{x}(k)$. This problem may be solved subject to constraints on both state and control variables (e.g. (6)) - a significant advantage of MPC over alternative control schemes.

In (2), the superscript $\cdot^*$ represents an optimised variable, and

$$\mathbf{x}^*(k+1:k+N_p) = \{\mathbf{x}^*(k+l|k) \mid l = \{1, 2, \ldots, N_p\}\} \quad (3)$$

is used as a shorthand notation for the sequence of optimal states over the prediction horizon. Both state and control sequences are used as optimisation variables in (2) to indicate that the direct collocation method [10] has been used. The optimisation of control states over the entire prediction horizon, $\mathbf{u}^*(k:k+N_p-1)$, is intentionally included. In contrast to alternative definitions (see, for example, [11]), the control horizon here refers to the number of discrete times at which the predicted control sequence is applied before (2) is solved again; i.e. $\mathbf{u}^*(k+1:k+N_c)$ are applied open-loop. This process is illustrated in Fig. 2.

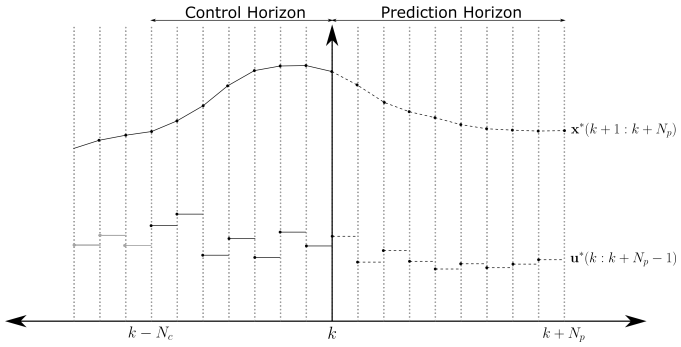


Fig. 2. Illustration of the iterative process of model predictive control

III. Methodology

A. Distributed Model Predictive Control Scheme

Predicted aircraft trajectories were generated by solving the following online optimisation problem over the prediction horizon N_p initialised at time k

$$\{\mathbf{x}_i^*(\cdot), \mathbf{u}_i^*(\cdot)\} = \arg\min J(\mathbf{x}_i(k+1:k+N_p), \mathbf{u}_i(k:k+N_p-1)), \quad (4)$$

where the objective function is

$$J(\mathbf{x}_i(\cdot), \mathbf{u}_i(\cdot)) = \sum_{l=1}^{N_p} \Bigg[\|\mathbf{x}^r(k+l) - \mathbf{x}_i(k+l|k)\|_Q^2 + \|\mathbf{u}_i(k+l-1|k)\|_R^2 + \sum_{j\in\mathcal{N}_i} \frac{1}{\|\mathbf{x}_i(k+l|k) - \mathbf{x}_j(k+l|k)\|_S^2 + \epsilon} + \sum_{j\in\mathcal{N}_i^v} \frac{1}{\|\mathbf{x}_i(k+l|k) - \mathbf{x}_j^v(k+l|k)\|_{S_v}^2 + \epsilon} \Bigg] \quad (5)$$

and the notation $\|\mathbf{x}\|_A^2 = \mathbf{x}^T A \mathbf{x}$ is used for brevity. In this formulation, the control horizon N_c refers to the number of predicted control inputs to be implemented before the optimisation problem is re-solved, $\mathbf{u}^*(k:k+N_c)$, in contrast with the alternative definition used elsewhere [11]. While this increases the number of optimisation variables to $(n+m)N_p$, rather than $nN_p + mN_c$, the approach was found to produce faster results with no detrimental effects provided detection distances and N_p/N_c were appropriately large. The traditional assumption of $\mathbf{u}^*(k+N_c+1:k+N_p) = \{\mathbf{0}\}$ was found to have detrimental effects for collision avoidance in constrained environments.

The aircraft's target point is represented by $\mathbf{x}^r$, and is assumed to be information pre-specified using a method such as distributed consensus [12]. The weighting matrices Q, R, S, S_v are assumed to be symmetric, and $\epsilon = 10^{-16}$ is used to promote collision avoidance between aircraft and obstacles. The set $\mathcal{N}_i$ represents the neighbouring aircraft of agent i, and $\mathcal{N}_i^v$ represents the neighbouring virtual agents, which provide repulsion from detected points on an obstacle. This coupling between aircraft in the optimisation problem above forms the distributed part of this algorithm, which is similar in form to that used in [8].

The optimal control problem is solved subject to the following constraints

$$\begin{cases} \mathbf{u}_i(k+l|k) \in \mathcal{U}_i \\ \mathbf{x}_i(k+l+1|k) \in \mathcal{X}_i \\ \mathbf{x}_i(k+l+1|k) = \mathbf{g}\left[\mathbf{x}_i(k+l|k), \mathbf{u}_i(k+l|k)\right] \end{cases} \quad (6)$$

for $l = \{0, 1, \ldots, N_p - 1\}$, where the set of admissable controls is defined as a standard box constraint

$$\mathcal{U}_i \triangleq \{\mathbf{u}_i \in \mathbb{R}^m \mid \mathbf{u}_{i,min} \leq \mathbf{u}_i \leq \mathbf{u}_{i,max}\} \quad (7)$$

and the set of feasible states is the region of state space excluding that occupied by the set of known obstacles $\mathcal{O}$ with centres $\mathbf{x}_c$ and dimensions defined in O_j (hereafter referred to as hard constraints).

$$\mathcal{X}_i \triangleq \{\mathbf{x}_i \in \mathbb{R}^n \mid 1 - \|\mathbf{x}_i(k+l|k) - \mathbf{x}_c\|_{O_j}^2 > 0 \quad \forall j \in \mathcal{O}, l = \{0, 1, \ldots, N_p\}\}. \quad (8)$$

To reduce the computational burden of solving in a constrained state space, the external active set method proposed in [13] is implemented.

The state generator function from (6) is defined using Euler's method

$$\mathbf{g}(\cdot) = \mathbf{x}_i(k+l|k) + \Delta t \cdot \mathbf{f}(\mathbf{x}(k+l|k), \mathbf{u}(k+l|k)) \quad (9)$$

where the aircraft's state is approximated using a Dubins path kinematic model [14]

$$\mathbf{x} = [x, y, z, \phi, \theta, \psi]^T \quad (10)$$

to reduce the number of state variables and thus decrease computation time.

The set of equations describing this kinematic model, $\mathbf{f}(\cdot)$, can be seen in (11) below, and are similar to the bicycle model used in similar work [15][16]. Notably, a constant airspeed V_T has been assumed in this work to both reduce the number of state variables and avoid the tendency of the optimisation algorithm to decrease the aircraft's speed for collision avoidance.

$$\begin{aligned} \dot{x} &= V_T \cos\theta \cos\psi \\ \dot{y} &= V_T \cos\theta \sin\psi \\ \dot{z} &= -V_T \sin\theta \\ \dot{\phi} &= u_{\dot{\phi}} \\ \dot{\theta} &= u_{\dot{\theta}} \\ \dot{\psi} &= \frac{g}{V_T} \tan\phi \end{aligned} \quad (11)$$

The optimisation algorithm used was COIN-OR's Interior Point Optimiser (IPOPT) [17], using the library's MATLAB interface and the linear solver MUMPS.

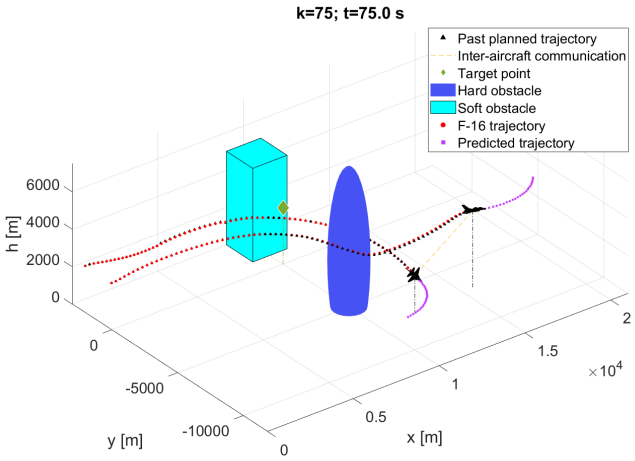


Fig. 3. Simulated F-16s capable of tracking MPC-generated trajectories over a simulated period of 75 s with $\Delta t = 1.0$ s

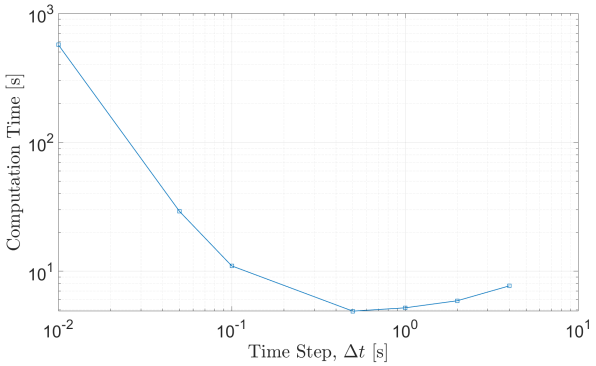


Fig. 4. Variation in computation time due to uniform time step

B. Nonlinear F-16 Model

Verification of the trajectories produced using the kinematic model described in (11) was performed using a nonlinear model of an F-16, created using data from [18][19]. Due to the increased complexity of this model, 13 state variables were required: highlighting the efficiency of the 6 kinematic states used in the MPC scheme. The state variables in (12) have their standard aerospace representations, with P_e included to represent the state of the F-16's engine power.

$$\hat{\mathbf{x}} = [U, V, W, P, Q, R, x, y, h, \phi, \theta, \psi, P_e]^T \qquad (12)$$

Tracking of the trajectories produce by (4) was performed using rudimentary PID attitude autopilots. These autopilots were simulated with a loop rate of 100 Hz, and were capable of tracking the MPC-generated trajectories (see Fig. 3) which were produced for $t = k\Delta t,\ \forall k = \{1, 2, \ldots, k_{max}\}$. Due to the simplicity of these control systems, the region of state space in which they were capable of stabilising the F-16 was relatively small; particularly unstable behaviour was observed during manoeuvres with simulateneous rapid pitch and roll rates. This limitation was addressed through the aforementioned control constraints on $u_{\dot{\phi}}, u_{\dot{\theta}}$ such that they were limited to ± 90 deg/s and ± 5 deg/s, respectively.

To reduce the steady-state tracking error between the F-16 and the MPC-predicted trajectories, the MPC scheme was reinitialised at the aircraft's position at the end of each control horizon, N_c. While this provided an effective means of improving tracking performance, it had undesirable effects for the use of state constraints, which is further discussed in Sec. IV-B.

IV. Results & Discussion

A. Parameter Selection

To establish an appropriate set of parameters for the distributed MPC scheme described above, two trade studies were performed. The first of these, illustrated in Fig. 4, examined the influence of the time step Δt on computation time. This data was obtained using a 1000 s, obstacle-free simulation of a single aircraft as a benchmark, with prediction horizon $T_p = N_p \Delta t = 20$ s and control horizon $T_c = N_c \Delta t = 4$ s. Fig. 4 demonstrates the drawbacks of excessively small time steps due to the significant number of optimisation variables required: $(n+m)T_p/\Delta t$. Conversely, increasingly large time steps were also observed to result in longer computation times. This is likely due to equality constraints (9) containing significant roundoff errors, and as such reducing the accuracy of the analytical derivatives provided to the solver. While these errors could be reduced through the use of a higher-order scheme, such as a Runge-Kutta method, the increased number of function evaluations may prove excessively burdensome.

When selecting a time step Δt, an important consideration is the distance $V_T \Delta t$ through which the aircraft travels without collision detection. To prevent predicted trajectories intersecting hard obstacles, it is advisable to select Δt such that this distance is significantly smaller than the minimum obstacle width. For this reason, obstacles were chosen with minimum widths of 4 km for $V_T \Delta t = 250$ m.

Selection of the control-prediction horizon ratio N_c/N_p was also examined, as can be seen in Fig. 5. This data illustrates the diminishing returns of increased control horizons, and motivates the use of $N_c/N_p \approx 0.1$. While larger values of N_c/N_p could be used for marginal reductions in computation time, this was observed to lead to obstacle collisions. This is due to the aircraft having insufficient time to manoeuvre due to effectively open-loop behaviour over the control horizon.

B. Collision Avoidance Using State Constraints

As can be seen in (8), elliptical equality constraints were used to define the feasible state set, $\mathcal{X}_i$. To reduce the significant computational load of these constraints, the external active set method proposed in [13] was used. In the simulation depicted in Fig. 6, this method was found to reduce the computational time by 34.6%, although this reduction is less prominent in problems with fewer state constraints.

A notable feature of the trajectories generated in Fig. 6 is their tangency to the obstacles. While this is appropriate to

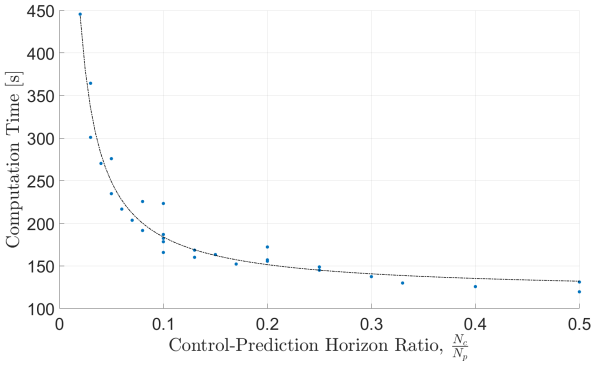


Fig. 5. Trade study between computation time and N_c/N_p

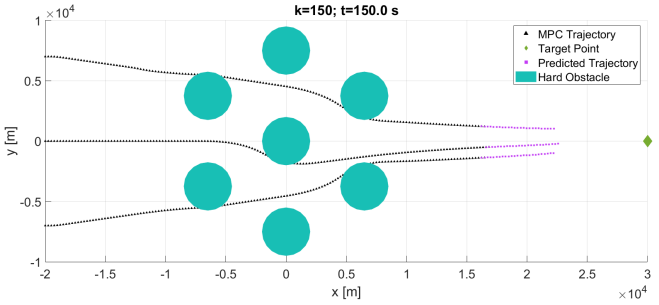


Fig. 6. Hard obstacle avoidance by MPC-generated trajectories over a 150 s simulation with $\Delta t = 1.0$ s

minimise control effort, it poses a problem when the MPC scheme is reinitialised at the aircraft's location, $\hat{\mathbf{x}}$. In cases where the aircraft's position tracking error leads it to pass into the hard obstacle, the MPC scheme is initialised at an infeasible point: $\mathbf{x}_i \notin \mathcal{X}_i$. As perfect trajectory tracking cannot be guaranteed, this presents a significant issue for this type of obstacle. While one method of mitigating this is the use of a repulsive virtual agent at the centre of these obstacles, this effectively reduces them to the objective function based approach to collision avoidance discussed below.

C. Collision Avoidance Using the Objective Function

A preferable form of collision avoidance to the state constraint approach is the use of terms in the objective function (4). Effective repulsion provided by the nearest detected points on obstacles $\mathcal{N}_i^v$, and neighbouring aircraft $\mathcal{N}_i$, allow the trajectory planner to begin avoiding collisions beyond the prediction horizon; as the limit now becomes the detection or inter-agent communication ranges. To maintain the individual impact of each of these points in the objective function, the maximum number of virtual agents must be limited. This can, under circumstances such as Fig. 7, lead to oscillatory trajectories. This highlights the need for a decision-making layer above the trajectory planner that can discern between obstacles to use a set of nearest detected points from each distinct obstacle.

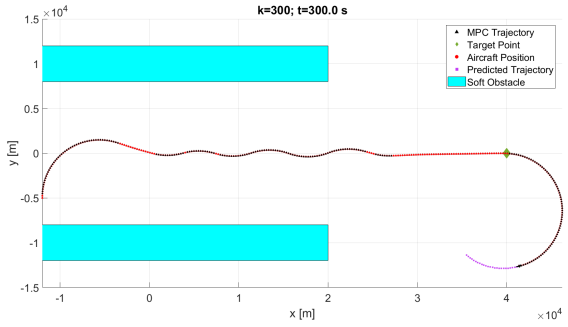


Fig. 7. Oscillating trajectory due to finite detection range and number of obstacle points, $|\mathcal{N}_i^v|$, captured after 300 s with $\Delta t = 1.0$ s

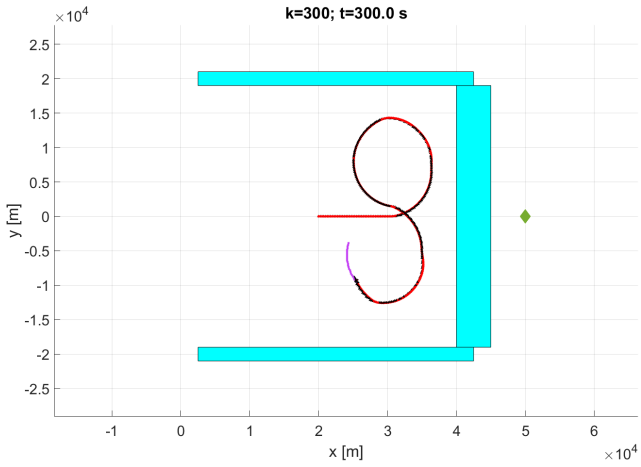


Fig. 8. Local minimum in the objective function formed by a dead-end obstacle with $\Delta t = 1.0$ s

While the oscillations shown above do not constitute a significant drawback of this approach, the potential existence of local minima do. Arrangements of obstacles such as those depicted in Fig. 8 produce a local minimum in the objective function, resulting in the trajectory planner becoming stuck in this region. While this could, in theory, be overcome by increasing N_p, there is no upper bound on what this value should be. Furthermore, the extreme computational cost of this approach make it undesirable. An alternative approach would be the use of a decision-making layer, as promoted above. This layer could shift the target point $\mathbf{x}^r$ to direct the aircraft away from the local minimum, and subsequently creating a virtual wall to prevent intrusion into the dead end. Nevertheless, in the absence of local minima, the use of objective function terms for collision avoidance has been found to work well; providing superior computational speed and robust performance relative to the state constraint approach.

D. Large-Scale Swarming

To demonstrate that the scheme proposed may be scaled to a significant number of swarming fixed-wing aircraft, a simulation of 32 aircraft was performed (see Fig. 9). Avoiding

the use of hard obstacles due to the aforementioned issues with state constraints, only soft obstacles have been included. In this simulation, the point of closest approach between any two aircraft is 676 m; a safe standoff distance for an F-16 with a 9 m wingspan. The two groups of 16 aircraft are shown to be able to 'squeeze' past an initial set of obstacles, before crossing perpendicular to each other without collisions. Motion in both lateral and vertical directions can be seen to be used for obstacle avoidance when necessary, with coordinated turns used preferentially due to the lower control cost.

V. Conclusion

A distributed MPC scheme has been proposed that is capable of producing realistic collision-free trajectory planning for fixed-wing aircraft. A nonlinear F-16 model has been implemented to verify that the kinematic model used to approximate fixed-wing dynamics is appropriate, and provides an effective means of reducing the number of state variables required to be optimised over the prediction horizon. An appropriate time step of $\Delta t \approx 1$ s has been identified for the F-16, and a ratio of $N_c/N_p \approx 0.1$ is shown through simulations to provide collision-free trajectories for a significant reduction in computational cost. A comparison between state constraints and objective function terms for collision avoidance has been made, and concluded that objective functions are superior. The potential presence of local minima was identified as a motivating reason for the implementation of a decision-making layer to direct the MPC trajectory planner in future work. A large-scale, collision-free, swarming simulation of 30 F-16 aircraft was presented to demonstrate this scheme's capabilities under adverse conditions.

Acknowledgment

The author would like to thank his supervisor, Dr. Hoam Chung, for his invaluable guidance and support throughout this project.

References

[1] Randal W Beard, Timothy W McLain, Derek B Nelson, et al. "Decentralized cooperative aerial surveillance using fixed-wing miniature UAVs". In: *Proceedings of the IEEE* 94.7 (2006), pp. 1306–1324.

[2] David W Casbeer, Derek B Kingston, Randal W Beard, et al. "Cooperative forest fire surveillance using a team of small unmanned air vehicles". In: *International Journal of Systems Science* 37.6 (2006), pp. 351–360.

[3] Daniel J Pack, Pedro DeLima, Gregory J Toussaint, et al. "Cooperative control of UAVs for localization of intermittently emitting mobile targets". In: *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)* 39.4 (2009), pp. 959–970.

[4] Jinlu Han, Yaojin Xu, Long Di, et al. "Low-cost multi-UAV technologies for contour mapping of nuclear radiation field". In: *Journal of Intelligent & Robotic Systems* 70.1-4 (2013), pp. 401–410.

[5] Achudhan Sivakumar and Colin Keng-Yan Tan. "UAV swarm coordination using cooperative control for establishing a wireless communications backbone". In: *Proceedings of the 9th International Conference on Autonomous Agents and Multiagent Systems: Volume 3-Volume 3*. International Foundation for Autonomous Agents and Multiagent Systems. 2010, pp. 1157–1164.

[6] Stephanie Petillo, Henrik Schmidt, and Arjuna Balasuriya. "Constructing a distributed AUV network for underwater plume-tracking operations". In: *International Journal of Distributed Sensor Networks* 8.1 (2011), p. 191235.

[7] Roy S Smith and Fred Y Hadaegh. "Control of deep-space formation-flying spacecraft; relative sensing and switched information". In: *Journal of Guidance, Control, and Dynamics* 28.1 (2005), pp. 106–114.

[8] David Shim, Hoam Chung, Hyoun Jin Kim, et al. "Autonomous exploration in unknown urban environments for unmanned aerial vehicles". In: *AIAA Guidance, Navigation, and Control Conference and Exhibit*. 2005, p. 6478.

[9] D Shim, H Kim, and S Sastry. "Decentralized reflective model predictive control of multiple flying robots in dynamic enviroment". In: *Department of Electical Engineering and Computer Sciences. University of California at Berkeley, Tech. Rep* (2003).

[10] Oskar Von Stryk and Roland Bulirsch. "Direct and indirect methods for trajectory optimization". In: *Annals of operations research* 37.1 (1992), pp. 357–373.

[11] Carlos E Garcia, David M Prett, and Manfred Morari. "Model predictive control: theory and practice—a survey". In: *Automatica* 25.3 (1989), pp. 335–348.

[12] Wei Ren and Randal W Beard. *Distributed consensus in multi-vehicle cooperative control*. Springer, 2008.

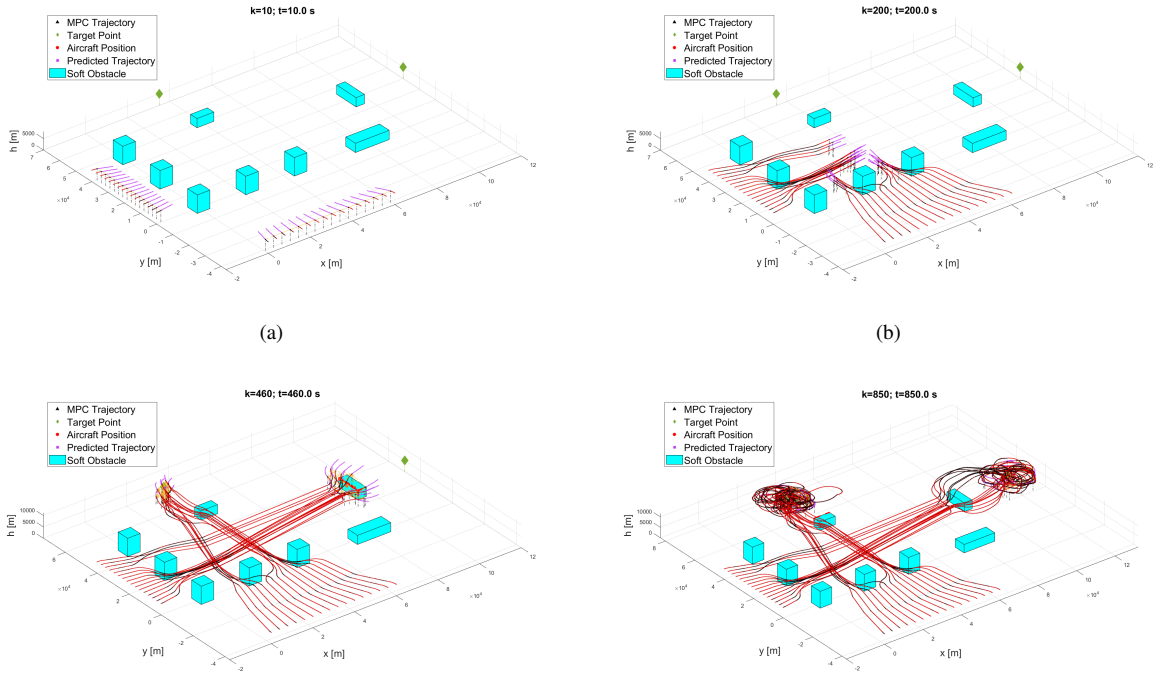
[13] Hoam Chung, Elijah Polak, and Shankar Sastry. "On the use of outer approximations as an external active set strategy". In: *Journal of optimization theory and applications* 146.1 (2010), pp. 51–75.

[14] Mark Owen, Randal W Beard, and Timothy W McLain. "Implementing dubins airplane paths on fixed-wing uavs". In: *Handbook of Unmanned Aerial Vehicles* (2015), pp. 1677–1701.

[15] Jessica Pannequin, Alexandre Bayen, Ian Mitchell, et al. "Multiple aircraft deconflicted path planning with weather avoidance constraints". In: *AIAA Guidance, Navigation and Control Conference and Exhibit*. 2007, p. 6588.

[16] Yeonsik Kang and J Karl Hedrick. "Linear tracking for a fixed-wing UAV using nonlinear model predictive control". In: *IEEE Transactions on Control Systems Technology* 17.5 (2009), pp. 1202–1210.

[17] Andreas Wächter and Lorenz T. Biegler. "On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming". In: *Mathematical Programming* 106.1 (2006), pp. 25–57. ISSN: 0025-5610. DOI: 10.1007/s10107-004-0559-y.



(a) (b)

(c) (d)

Fig. 9. Swarming of 32 F-16 aircraft in the presence of obstacles, with snapshots at four points in time using $\Delta t = 1.0$ s, $N_p = 25$, and $N_c = 3$

[18] Eugene A Morelli. "Global nonlinear parametric modelling with application to F-16 aerodynamics". In: *Proceedings of the 1998 American Control Conference. ACC (IEEE Cat. No. 98CH36207)*. Vol. 2. IEEE. 1998, pp. 997–1001.

[19] Brian L Stevens, Frank L Lewis, and Eric N Johnson. *Aircraft control and simulation: dynamics, controls design, and autonomous systems*. John Wiley & Sons, 2015.