A Jamming Game for Fleets of Mobile Vehicles

Walton P. Coutinho* Jörg Fliege†

October 26, 2024

Abstract

We consider a two-player Nash game in which each player represents a fleet of unmanned aerial vehicles. Each fleet is supposed to distribute information among fleet members, while simultaneously trying to prevent the opposite fleet from achieving this. Using the electromagnetic spectrum's properties, we model each fleet's task as a nonlinear Nash game. By reformulating conditions for Nash equilibria, we provide a fixed-point algorithm for this game and show its convergence under mild conditions. In practice, the game considered will often be solved in a distributed setting, where each member of each fleet computes information separately. We thus propose a corresponding distributed optimization algorithm and show its convergence. Numerical results illustrate the flexibility of our approach.

Keywords: drones, unmanned aerial vehicles, jamming, electronic warfare, Nash games

1 Introduction

Electronic Warfare (EW) is a military domain that uses the electromagnetic spectrum to gain an advantage in various operational environments. The electromagnetic spectrum encompasses a range of frequencies, including radio waves, microwaves, infrared and visible light, among others. An electronic attack involves the use of electromagnetic energy to disrupt, deceive, or destroy the functionality of electronic systems, such as radars, communication systems, or sensors. In particular, *jamming* aims to interfere with an adversary's ability to collect and disseminate information. Communications jamming is directed at disrupting radio communication systems by transmitting interfering signals on the same frequency as the targeted communication system, causing interference and making it difficult for the intended message to be received.

Unmanned aerial vehicles (UAVs or drones) and electronic warfare represent critical components of modern military and security strategies. Drones have revolutionized reconnaissance, surveillance, and strike capabilities. They provide real-time data, enhance situational awareness, and reduce the risk to human operators. In the context of electronic warfare, drones play a significant role both as platforms for electronic warfare systems and as potential targets. Drones equipped with electronic warfare capabilities can disrupt or manipulate communication systems, radar signals, and other

*Universidade Federal de Pernambuco, Programa de Pós-graduação em Engenharia de Produção do Centro Acadêmico do Agreste (PPGEP-CAA), Walton.Coutinho@ufpe.br

†Universidade Federal de Pernambuco, Programa de Pós-graduação em Engenharia de Produção do Centro Acadêmico do Agreste (PPGEP-CAA), University of Southampton, School of Mathematical Sciences, J.Fliege@soton.ac.uk

electronic devices, offering a powerful tool for both offensive and defensive actors. Nonetheless, drones are also susceptible to electronic warfare measures themselves: anti-drone systems can use such electronic warfare techniques to jam or spoof the communication links between a drone, its operator, and other units.

In this work, we consider fleets of UAVs that are antagonistic to each other. Members or agents of each fleet are tasked with sharing information with other members of their fleet, i. e., they want to send and receive data among each other. However, they also have the option to degrade or interfere with the information flow between members of the other fleet. Of crucial importance in this model is the ability of each agent to be mobile, i. e., to spend some of its energy to move around in space. By moving closer to an agent of the same fleet, information exchange between these two agents is improved. By moving closer to agents of the opposite fleet, interrupting the information flow of the opposition (jamming) is ameliorated.

We model this problem as a novel Nash game, where both players control their relevant fleets and are aware of the status and location of their agents and the location of the agents of the opposition. For each agent, decision variables include the power expenditure to communicate with other agents of the same fleet, power expenditure to jam communication between pairs of agents in the opposite fleet, and power expenditure and direction of any movement. Previous attempts at modeling this problem either use stationary agents only (see, e. g., Khanafer, Bhattacharya, Bas, et al. (2011)) or allow only one jamming agent to move (see Bhattacharya and Başar (2011a)). Another generalization included in our model is the consideration of communication and jamming graphs, which can be used to represent different strategic configurations and capabilities of the different UAVs in their fleets. Our communication and jamming model reflects the underlying physical reality of communication within the electromagnetic spectrum and contains no linearizations.

To find an equilibrium point of our game, we use a well-known fixed-point reformulation for Nash games (Rosen (1965)). Despite the lack of convexity, we show its convergence for a slightly modified problem formulation, where we essentially prevent UAVs from moving too close to each other. Next, motivated by a distributed computing approach, we provide a block-descent algorithm for the subproblems occurring in our fixed-point formulation and show its convergence under mild conditions. Extensive numerical experiments show the efficacy of our approach.

Our model follows the work of Bhattacharya, Khanafer, and Başar (2016), Khanafer, Bhattacharya, Bas, et al. (2011), and Khanafer, Bhattacharya, and Başar (2011), who considered a simplified version of our Nash game with two fleets of stationary agents, each fleet consisting of exactly two agents. A. Gupta et al. (2012) consider a game with incomplete information in which one player tries to jam the connection between a transmitter and a receiver. Bhattacharya and Başar (2012) consider teams of agents able to move in the plane, but where movement itself incurs no cost, and it is unclear how power allocation takes place. A scenario in which one mobile intruder jams the communication network of a formation of several vehicles, either all ground-based or all aerial, is considered in Bhattacharya and Başar (2010, 2011b, 2013), Bhattacharya, A. K. Gupta, and Başar (2013), and Parras, Zazo, et al. (2017). In particular, Parras, Val, et al. (2016) describe a stochastic version of this scenario. Strategies and counter-strategies for jamming vehicular ad-hoc networks have been described by Lu et al. (2017). Recently, a Markov game formulation for a jamming game has been considered by Feng et al. (2023), while tasks assignment problems for jamming swarms have been cast in game-theoretic form by Zhang et al. (2023).

Practically all formulations discussed above lead to Nash games; in contrast, Stackelberg games are only considered by a few authors, see e. g., Liu et al. (2024), Su et al. (2024), and Z. Yin et al. (2024) for some recent approaches. Other game-theoretic models on jamming UAVs are discussed

in Fu et al. (2023), Lv et al. (2017), J. Xu et al. (2021), and Yifan Xu et al. (2018), while Mkiramweni et al. (2019) and Li et al. (2024) provide surveys of game-theoretic approaches for jamming wireless networks.

The rest of this paper is organised as follows. In Section 2, we first derive a game between two players, each operating a fleet of UAVs, where each player has to solve a biobjective optimization problem with convex objectives, conic constraints, and nonconvex constraints, where the nonconvexity stems either from certain bilinear terms or from a strictly concave function. After a suitable scalarization, we arrive at the zero-sum Nash game we focus on further. Due to the highly nonlinear nature of this game, standard tools like reformulating the game into an MPEC do not work in this setting. Moreover, we need to expect that any solution method will be used in a distributed fashion, i. e., by the swarm of UAVs themselves. Section 3 therefore considers specific numerical solution methods for this game, together with corresponding convergence proofs. We report on the computational performance of our approach in Section 4, while Section 5 provides a discussion of our results and pointers to future research.

2 The Model

Suppose we have given two finite nonempty index sets F and G, with $F \cap G = \emptyset$, denoting two fleets of agents or UAVs. We use indices $i, j, k, \ell \in F \cup G$ to indicate agents, and will usually use the convention $i, j \in F$ and $k, \ell \in G$, unless explicitly stated otherwise. For each $i \in F$, $k \in G$, we have been given locations $x_i^0, x_k^0 \in \mathbb{R}^2$ and a maximum amount of energy to spend, P_i^{max}, P_k^{max} (usually measured in Watts). Furthermore, we have given two edge sets $E(F) \subseteq F \times F$ and $E(G) \subseteq G \times G$, indicating which agent needs to communicate with which other agent. In other words, if $(i, j) \in E(F)$, then agent i needs to send some data to agent j. We assume without loss of generality that $(i, i) \notin E(F)$ for all $i \in F$ and $(k, k) \notin E(G)$ for all $k \in G$.

Suppose that we have $e = (i, j) \in E(F)$ and consider different time slots $t = 1, \ldots, T$ for communication. Following Khanafer, Bhattacharya, Bas, et al. (2011), if the UAV i uses $p_{e,t}$ amount of power for this communication (i. e., it sends a signal of strength $p_{e,t}$), then the UAV j receives an amount of power $p_{e,t}^R$ equal to

$$p_{e,t}^R = \varrho p_{e,t} d_{e,t}^{-\alpha}. \tag{1}$$

Here, $d_{e,t} = d_{i,j,t} = \|x_{i,t} - x_{j,t}\|$ is the Euclidean distance between agents i and j, which at time t are located at $x_{i,t}$ and $x_{j,t}$, respectively. In the above and in what follows, we always assume $x_{i,t} \neq x_{j,t}$, and in all formulations of optimization problems we will have constraints ensuring that $p_{e,t}^R$ does not grow beyond reasonable bounds. The constant α is the so-called *path-loss coefficient* that depends on the environment (atmospheric conditions, terrain, etc.) in which i and j try to communicate and is usually chosen between 2 and 4. The constant ϱ, the *antenna gain*, depends on the design of the antennas of i and j and the signal wavelength in use; in what follows we will assume that ϱ does not depend on i and j.

A crucial quantity for the communication efforts along the edge $e = (i, j)$ is the signal-to-interference-and-noise ratio $SINR_{e,t}$, which is defined as

$$\mathrm{SINR}_{e,t} = \frac{p_{e,t}^R}{\sigma^2 + I_{e,t}}, \qquad (e \in E(F); t = 1, \ldots T). \tag{2}$$

Here, σ^2 represents background noise due to, for instance, solar radiation. This is usually a random variable with Gaussian distribution. However, its expected value is usually several orders of magnitude smaller than any of the powers used by any of the agents, and can thus be approximated by a constant. Furthermore, $I_{e,t}$ is the total interference power along the edge e due to jamming from the enemy fleet. Suppose that for each agent k from the opposite team we have given information as to if k is capable, in principle, of jamming communications along edge e at time t. Let $J(e) \subseteq G$ be the set of opposite agents able to jam communications along edge $e \in E(F)$, and denote by $p^J_{k,e,t}$ the power used to jam communications along edge e at time t by some $k \in J(e)$. Then we have

$$I_{e,t} = \sum_{k \in J(e)} \varrho p^J_{k,e,t} d^{-\alpha}_{k,j,t}, \qquad (j \in F, e = (i,j) \in E(F)) \tag{3}$$

with the same constants as before, $d_{k,j,t} = \|x_{k,t} - x_{j,t}\|$, and the summation is over all agents of the opposite team. In total, we have

$$\mathrm{SINR}_{e,t} = \frac{\varrho p_{e,t} d^{-\alpha}_{e,t}}{\sigma^2 + \varrho \sum_{k \in J(e)} p^J_{k,e,t} d^{-\alpha}_{k,j,t}}, \qquad (e = (i,j) \in E(F); t = 1, \ldots, T). \tag{4}$$

Figure 1 illustrates this concept in the plane. Here, we have given three agents a, b, and c, located at coordinates $(2,2)$, $(2,8)$ and $(8,8)$, respectively. We assume that they all communicate simultaneously with some fourth UAV. Here, we have used $\alpha = 2$. Figure 1 (a) shows a heatmap of the sum of all three SINRs that the fourth UAV experiences, depending on its location in the plane. As it can be clearly seen, it is beneficial for the fourth agent to be as close to one of the other agents as possible. Figure 1 (b) considers the effect of jamming. Here we have one adversarial agent located at $(2.5, 8)$, using a jamming power equal to the agent's b communication power, thus decreasing the SINR between agent b and the fourth UAV. One can observe that the overall contribution of agent b to the sum of SINRs as experienced by the fourth UAV is greatly diminished.

In practice, it is these quantities $\mathrm{SINR}_{e,t}$ that the fleet F wants to maximise, as these are closely related to various performance indices used in wireless communications. For example, consider the case in which agent i communicates digital data, i. e., individual bits. Of relevance is then the bit error rate $\mathrm{BER}_{e,t}$, the average number of bits received by agent j along $e = (i,j)$ which are received incorrectly. It can then be shown (Palomar, Bengtsson, and Ottersten (2005)) that in many cases we have

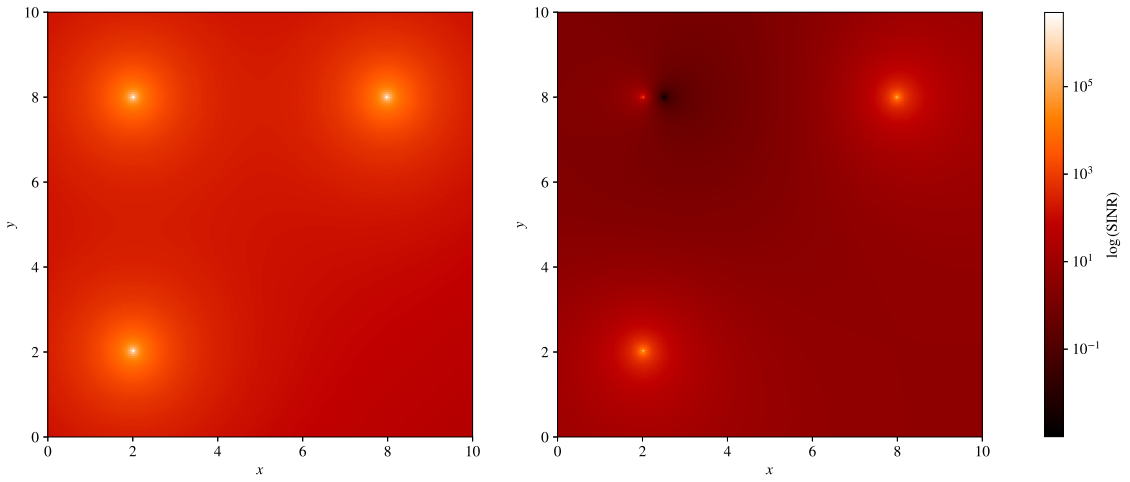
$$\mathrm{BER}_{e,t} = \mathrm{const}_1 Q(\mathrm{const}_2 \sqrt{\mathrm{SINR}_{e,t}}), \qquad (e \in E(F); t = 1, \ldots, T) \tag{5}$$

with known positive constants and Q the tail probability of the Gaussian distribution. In other words, $\mathrm{BER}_{e,t}$ is monotonically decreasing in $\mathrm{SINR}_{e,t}$.

In total, the *first* objective of the fleet of agents F is thus

$$\max \sum_{t=1}^{T} \sum_{e=(i,j) \in E(F)} \frac{\varrho p_{e,t} d^{-\alpha}_{e,t}}{\sigma^2 + \varrho \sum_{k \in J(e)} p^J_{k,e,t} d^{-\alpha}_{k,j,t}}, \tag{6}$$

where the decision variables are the power levels $p_{e,t}$ $(e \in E(F);\ t = 1, \ldots, T)$ and the locations $x_{i,t}, x_{j,t} \in \mathbb{R}^2$ determining $d_{i,j,t} = \|x_{i,t} - x_{j,t}\|$, $d_{k,j,t} = \|x_{k,t} - x_{j,t}\|$. Note that the power levels



(a) Three agents communicating

(b) Communication in the presence of a jamming adversary

Figure 1: Total SINR as experienced by a UAV in the plane, with three stationary agents communicating with the UAV from positions $(2, 2)$, $(2, 8)$ and $(8, 8)$. Figure (a) shows the case without jamming, Figure (b) depicts the case of one adversary agent placed at $(2.5, 8)$, jamming the UAV at $(2, 8)$.

$p^J_{k,e,t}$ are decision variables of the opposite fleet with agents indexed by k. In turn, fleet F controls decision variables $p^J_{j,k,t}$ ($j \in F, k \notin F$; $t = 1, \ldots, T$) that show up in the corresponding objective of the opposite fleet.

However, note that in this objective the effect of jamming agents from the opposite fleet is not taken into account. We therefore consider the first objective of the opposite fleet, which the opposite fleet wants to maximize, as an objective to be minimized by fleet F. Using the same notational conventions as before, with the roles of fleets F and G interchanged, the *second* objective of the fleet F thus becomes

$$\min \sum_{t=1}^{T} \sum_{e=(k,\ell)\in E(G)} \frac{\varrho p_{e,t} d_{e,t}^{-\alpha}}{\sigma^2 + \varrho \sum_{i\in J(e)} p^J_{i,e,t} d^{-\alpha}_{i,\ell,t}} \tag{7}$$

Given that the agents are mobile, we have that their locations $x_{i,t}, x_{j,t}, x_{k,t}$ are decision variables as well. We thus assume that each agent i has a starting position $x_{i,0} \in \mathbb{R}^2$. At each time step t, the agent i first moves from $x_{i,t-1}$ to $x_{i,t}$ before sending or jamming. The movement results in an energy expenditure of $c_i \|x_{i,t} - x_{i,t-1}\|$, where c_i is a conversion factor of dimension W/m. We can then write the two objectives of fleet F as one bi-objective optimization problem

$$\max_{p_{e,t}, p^J_{i,k,t}, x_{i,t}} \begin{bmatrix} \sum_{t=1}^{T} \sum_{e=(i,j)\in E(F)} \frac{\varrho p_{e,t} \|x_{i,t} - x_{j,t}\|^{-\alpha}}{\sigma^2 + \varrho \sum_{k\in J(e)} p^J_{k,e,t} \|x_{k,t} - x_{j,t}\|^{-\alpha}} \\ -\sum_{t=1}^{T} \sum_{e=(k,\ell)\in E(G)} \frac{\varrho p_{e,t} \|x_{k,t} - x_{\ell,t}\|^{-\alpha}}{\sigma^2 + \varrho \sum_{i\in J(e)} p^J_{i,e,t} \|x_{i,t} - x_{\ell,t}\|^{-\alpha}} \end{bmatrix}, \tag{8}$$

with decision variables

- $p_{e,t}$ ($e = (i,j) \in E(F)$, $i \neq j$; $t = 1, \ldots, T$) power used to communicate from i to j at time t.
- $p^J_{i,e,t}$ ($i \in F$, $e = (k,\ell) \in E(G)$; $t = 1, \ldots, T$) power used to jam the communication of opposite agents $k, \ell \in G$ along edge $e = (k,\ell) \in E(G)$ at time t.
- $x_{i,t}$ ($i \in F$; $t = 1, \ldots, T$) location of agent i at time t, after movement and before communication and jamming takes place.

As constraints, we have the *power constraint* for each agent i,

$$\sum_{t=1}^{T} \left(\sum_{e=(i,j)\in E(F)} p_{e,t} + \sum_{e \in E(G)} p^J_{i,e,t} + c_i \|x_{i,t} - x_{i,t-1}\| \right) \leq P_i^{max} \qquad (i \in F), \tag{9}$$

a constraint on the movement of each agent i, essentially providing a *velocity constraint*,

$$\|x_{i,t} - x_{i,t-1}\| \leq \delta_i \qquad (i \in F;\ t = 1, \ldots, T), \tag{10}$$

with $\delta_i > 0$ some given parameter and the starting position $x_{i,0} \in \mathbb{R}^2$ for $t = 0$ predefined, a *collision avoidance constraint*,

$$\|x_{i,t} - x_{j,t}\| \geq \epsilon \qquad (i, j \in F \cup G, i \neq j; t = 1, \ldots, T) \tag{11}$$

with some $\epsilon > 0$, and finally *nonnegativity constraints*

$$p_{e,t}, p^J_{i,\bar{e},t} \geq 0 \qquad (e \in E(F), i \in F, \bar{e} \in E(G); t = 1, \ldots, T). \tag{12}$$

Further constraints on x_i can indicate geographical constraints, constraints from modeling the flight dynamics, mission constraints, etc.

In what follows we will assume that the set of feasible points is nonempty for all $x_{j,t} \in \mathbb{R}^2$ ($j \in G$, $t = 1, \ldots, T$). These $x_{j,t}$ occur only in the collision avoidance constraint. In essence, we assume here that ϵ occurring in (11) is so small that the points in space that each agent of fleet F can reach cannot be covered by $|F \cup G|$ balls of radius ϵ.

Remark. Without loss of generality, we can assume that the power constraint (9) will be active, as the first objective is strictly monotone in the $p_{e,t}$ and the second objective is strictly monotone in the $p^J_{i,e,t}$. In other words, if we are at a point where (9) is not active, we can increase the value of a $p_{e,t}$ or a $p^J_{i,e,t}$ by a small amount and arrive at a point that strictly dominates the original one in the bi-objective problem defined above, in the sense of Pareto.

Remark. For the same reason as in the remark above, we have formally introduced decision variables $p^J_{i,\bar{e},t}$ for $i \in F$ and all $\bar{e} \in E(G)$, and not only for those i with $i \in J(\bar{e})$. Due to the monotonicity of both objectives, we will have $p^J_{i,\bar{e},t} = 0$ for all $i \notin J(\bar{e})$ at optimality. Of course, any optimization tool can be supported in its search by adding these conditions as constraints.

Remark. If an agent has additional known energy expenditure $\bar{P}_{i,t} \geq 0$ at time t, we can adapt (9) and modify the right-hand side to $P_i^{max} - \sum_{t=1}^{T} \bar{P}_{i,t}$. In case of a priori unknown additional energy expenditures, we would need to include additional integer or binary variables $y_{i,t}$ to the model and add a term of the form $\sum_{t=1}^{T} \bar{P}_{i,t} y_{i,t}$ to the left-hand side of (9).

Remark. Only the collision avoidance constraints are nonconvex (in fact, concave), and these are the only constraints coupling decision variables from different fleets with each other.

For ease of notation, denote now by x the vector of decision variables of fleet F (consisting of variables $p_{e,t}, p^J_{i,\bar{e},t}, x_{i,t}$ for indices $e \in E(F), i \in F, \bar{e} \in E(G)$, and $t = 1, \ldots, T$) and by y the vector of decision variables of the opposite fleet (consisting of variables $p_{e,t}, p^J_{k,\bar{e},t}, x_{k,t}$ for indices $e \in E(G), k \in G, \bar{e} \in E(F)$, and $t = 1, \ldots, T$). We can then write the two objectives involved as

$$f_1(x,y) = \sum_{t=1}^{T} \sum_{e=(i,j)\in E(F)} \frac{\varrho p_{e,t} \|x_{i,t} - x_{j,t}\|^{-\alpha}}{\sigma^2 + \varrho \sum_{k\in J(e)} p^J_{k,e,t} \|x_{k,t} - x_{j,t}\|^{-\alpha}} \tag{13}$$

and

$$f_2(x,y) = \sum_{t=1}^{T} \sum_{e=(k,\ell)\in E(G)} \frac{\varrho p_{e,t} \|x_{k,t} - x_{\ell,t}\|^{-\alpha}}{\sigma^2 + \varrho \sum_{i\in J(e)} p^J_{i,e,t} \|x_{i,t} - x_{\ell,t}\|^{-\alpha}} \tag{14}$$

and the bi-objective problem of fleet F as

$$\max_{x\in C_F(y)} \begin{bmatrix} f_1(x,y) \\ -f_2(x,y) \end{bmatrix}, \tag{15}$$

parameterized in y, where $C_F(y)$ denotes the set of feasible points of fleet F. Then, the opposite fleet faces the problem

$$\max_{y\in C_G(x)} \begin{bmatrix} f_2(x,y) \\ -f_1(x,y) \end{bmatrix}, \tag{16}$$

parameterized in x, with $C_G(x)$ being the set of feasible y.

A typical strategy to deal with multiobjective problems is *scalarization*. Here, each fleets F, G chooses a parameter $\omega_F, \omega_G \in [0,1]$. Fleet F then replaces the bi-objective problem above by

$$\max_{x\in C_F(y)} \omega_F f_1(x,y) - (1-\omega_F) f_2(x,y), \tag{17}$$

and fleet G does likewise. The two parameters ω_F, ω_G represent the importance of the two objectives for the two fleets. Denote now by θ_F and θ_G the objectives of both fleets F and G in minimization form, i. e., $\theta_F = -\omega_F f_1 + (1-\omega_F) f_2$ and $\theta_G = -\omega_G f_2 + (1-\omega_G) f_2$. The two fleets then face the problems

$$\min_{x\in C_F(y)} \theta_F(x,y) \tag{18}$$

and

$$\min_{y \in C_G(x)} \theta_G(x, y). \tag{19}$$

One can observe that for $\omega_F = \omega_G = 1/2$ the objectives of the two fleets are essentially the same; they just 'swap' the variables x and y around and the game becomes a *zero-sum Nash game*. In the general case, we are looking for an equilibrium, i. e., a pair of feasible strategies x^*, y^* with $x^* \in C_F(y^*)$, $y^* \in C_F(x^*)$, for which we have

$$\begin{aligned} \theta_F(x^*, y^*) &\leq \theta_F(x, y^*) \qquad \forall x \in C_F(y^*), \\ \theta_G(x^*, y^*) &\leq \theta_G(x^*, y) \qquad \forall y \in C_G(x^*). \end{aligned}$$

In other words, we want to find a solution x^* to problem (18) where we use the parameter $y = y^*$, and a solution y^* to problem (19) where we use the parameter $x = x^*$.

The existence of such equilibria is usually only confirmed for convex Nash games, i. e., the sets C_F and C_G are assumed to be convex and compact, and f_1, f_2 are assumed to be smooth. These are of course the standard conditions for Brouwer's fixed point theorem. In our case, only compactness of C_F and C_G hold. However, the nonconvexity in the constraints is concentrated in the collision avoidance constraints $\|x_{i,t} - x_{j,t}\| \geq \epsilon > 0$. For appropriately chosen problem parameters P_i^{max}, c_i, δ_i and ϵ the sets $C_F(y)$ and $C_G(x)$ are nonempty and compact and there exists points $x^{ref} \in C_F(y^{ref})$, $y^{ref} \in C_G(x^{ref})$ from the interior of those sets. It is then straightforward to show that the assumptions (A), (B), (C) of Pang and Scutari (2011) hold, which assures via their Theorem 5 the existence of a quasi-Nash equilibrium, i. e., a solution to the variational inequality associated with our game.

In the next section, we will provide a stronger result for a slightly modified version of our problem. Namely, we will show that a Nash equilibrium always exists and that a specific fixed-point iteration converges to such an equilibrium under some mild conditions.

3 Numerical Solution Methods

Since all functions involved are smooth, we can characterise (local) equilibria by the KKT conditions for these optimization problems. To solve such conditions numerically, one often resorts to an MPEC (Mathematical Program with Equilibrium Constraint) solver. However, preliminary numerical tests with state-of-the-art MPEC solvers have shown that this approach does not seem to be numerically feasible in our case, as all such solvers get stuck at infeasible points. In Subsection 3.1, we will therefore provide an alternative numerical approach, based on a natural fixed-point formulation of the problem.

Moreover, in a practical setting, we need to expect that any such solution method will be used in a distributed fashion, i. e., by the swarm of UAVs under consideration, which each UAV being its own computational unit. Subsection 3.2 provides us with a distributed algorithm for the fixed-point method described in Subsection 3.1.

3.1 A Fixed-Point Formulation

For a given $(\hat{x}, \hat{y})$ let us consider the optimization problem

$$\min_{x,y} \quad \theta_F(x, \hat{y}) + \theta_G(\hat{x}, y) \tag{20}$$

$$\text{subject to} \quad x \in C_F(\hat{y}), y \in C_F(\hat{x}). \tag{21}$$

Suppose that (x^*, y^*) is a solution to this problem. It is well known (see Rosen (1965)) that (x^*, y^*) is a Nash equilibrium if $(x^*, y^*) = (\hat{x}, \hat{y})$, i. e., if (x^*, y^*) is a fixed point of the solution mapping of (20)–(21).

A standard approach to circumvent numerical issues related to the nonuniqueness of any optimum of (20) is to add a regularisation term and consider

$$\min_{x,y} \quad \theta_F(x, \hat{y}) + \theta_G(\hat{x}, y) + r(\|x - \hat{x}\|^2 + \|y - \hat{y}\|^2) \tag{22}$$

$$\text{subject to} \quad x \in C_F(y), y \in C_G(x) \tag{23}$$

with a parameter $r > 0$.

The following algorithm provides a fixed-point iteration for our Nash game:

1. Choose $r > 0$ and $\hat{x}, \hat{y}$ such that $\hat{x} \in C_F(\hat{y})$ and $\hat{y} \in C_G(\hat{x})$.
2. Solve (22)–(23). Denote the result by (x, y).
3. Set $(\hat{x}, \hat{y}) := (x, y)$.
4. Update r.
5. Goto step 2.

Step 2 of the algorithm can obviously be split up into solving the two problems

$$\min_{x \in C_F(\hat{y})} \quad \theta_F(x, \hat{y}) + r\|x - \hat{x}\|^2, \tag{24}$$

$$\min_{y \in C_G(\hat{x})} \quad \theta_G(\hat{x}, y) + r\|y - \hat{y}\|^2, \tag{25}$$

which can be solved in parallel. Both correspond to "best response" optimization problems, i. e., (24) computes the strategy of fleet F if it is known that fleet G will deploy strategy $\hat{y}$. The overall algorithm is thus one of successive improvements in strategies x and y.

A convergence proof for the case of such a fixed-point algorithm applied to convex problems can be found in Flåm and Antipin (1996), who also describe various adaptations like solving only for ε-minima in each step, appropriate strategies for changing the parameter r, etc. Regarding the convergence speed, we only have $\sum_k \|z_k - z_{k+1}\|^2 < +\infty$ from Flåm and Antipin (1996) for the sequence of iterates $z_k = (x_k, y_k)$ of the algorithm, leading only to the somewhat weak statement $\|z_k - z_{k+1}\| = O(1/\sqrt{k})$. Note also that for nonconvex problems as we are facing them here, the fixed-point scheme *without* regularization term can cycle or stagnate. To prove the convergence of our method, we need to modify the problem under consideration ever so slightly as follows.

Denote by $\tilde{C}_F$ and $\tilde{C}_G$ the sets of feasible points for player F and player G without the collision avoidance constraints (11). Note that $C_F(y) \subset \tilde{C}_F$, $C_G(x) \subset \tilde{C}_G$ and $\text{conv}(C_F(y)) = \tilde{C}_F$,

$\text{conv}(C_G(x)) = \tilde{C}_G$. In other words, $\tilde{C}_F$ and $\tilde{C}_G$ are the smallest convex sets that contain $C_F(y)$ and $C_G(x)$. Next, define objective functions $\tilde{f}_1$ and $\tilde{f}_2$ on $\tilde{C}$ as follows. We set $\tilde{f}_1(x,y) = f_1(x,y)$ and $\tilde{f}_2(x,y) = f_2(x,y)$ if $x \in C_F(y)$ and $y \in C_G(y)$. Otherwise, there are some indices $i, j \in F \cup G$, $i \neq j$ and some t such that $\|x_{i,t} - x_{j,t}\| < \epsilon$. In the formula for $f_1(x,y)$ and $f_2(x,y)$ we then replace each occurrence of $\|x_{i,t} - x_{j,t}\|$ by the parameter $\epsilon > 0$. This is equivalent to specifying that in the definition of f_1 and f_2 we replace each term $\|x_{i,t} - x_{j,t}\|$ by $\max\{\epsilon, \|x_{i,t} - x_{j,t}\|\}$ and call the resulting functions $\tilde{f}_1$ and $\tilde{f}_2$. Intuitively, this should not change the structure of the solution: there is no incentive for two agents $i, j \in F$ to decrease their distance at time t below ϵ, as there will be no increase in received signal strength. The same can be said for jamming strengths.

Note that both $\tilde{f}_1$ and $\tilde{f}_2$ are continuous but not necessarily differentiable at $\|x_{i,t} - x_{j,t}\| = \epsilon$. It is straightforward to modify the definitions of $\tilde{f}_1$ and $\tilde{f}_2$ in such a way that for $\delta < \epsilon$ we replace $\|x_{i,t} - x_{j,t}\|$ with $\epsilon - \delta$ if $\|x_{i,t} - x_{j,t}\| \leq \epsilon - \delta$, and we provide a smooth continuation of the term in the region $\epsilon - \delta \|x_{i,t} - x_{j,t}\| < \epsilon$. This does not affect the convergence result below, but it might help any algorithmic scheme relying on the smoothness of objective functions.

Let us now set $\tilde{\theta}_F = -\tilde{f}_1 + \tilde{f}_2$ and $\tilde{\theta}_G = -\tilde{\theta}_F$ and consider

$$\min_{x,y} \quad \tilde{\theta}_F(x, \hat{y}) + \tilde{\theta}_G(\hat{x}, y) + r(\|x - \hat{x}\|^2 + \|y - \hat{y}\|^2) \tag{26}$$

$$\text{subject to} \quad x \in \tilde{C}_F, y \in \tilde{C}_G \tag{27}$$

with a regularization parameter $r > 0$.

To show convergence, we will make use of the following concept. Define the compact set $X := \tilde{C}_F \times \tilde{C}_G$ and consider the bifunction $f : X \times X \longrightarrow \mathbb{R}$ defined as follows. Let $a = (x_a, x_b) \in X$ and $b = (x_b, y_b) \in X$. Then we set $f(a,b) = \tilde{\theta}_F(x_a, y_b) + \tilde{\theta}_G(x_b, y_a)$. It is easy to see that $f(a, \cdot)$ is uniform Lipschitz for all $a \in X$. Let us denote the corresponding Lipschitz constant by L.

Theorem. *Consider the fixed point iteration with (22)–(23) replaced by (26)–(27). Let $(r_k)_k$ be the sequence of regularization parameters used in this fixed-point iteration, and let $(x_k, y_k)_k$ be the sequence generated by this iteration.*

- *Assume that $r_k \geq r_U$ for some constant $r_U > 0$ for all k and that in each iteration we compute a global minimum of $\tilde{\theta}_F(\cdot, \hat{y})$ and $\tilde{\theta}_G(\hat{x}, \cdot)$. Assume that we have $\|(x_{k+1}, y_{k+1}) - (x_k, y_k)\| \longrightarrow 0$. Then, $(x_k, y_k)_k$ converges to a Nash equilibrium of the modified game with objectives $\tilde{\theta}_F$, $\tilde{\theta}_G$ and feasible sets $\tilde{C}_F$ and $\tilde{C}_G$.*
- *Suppose that the sequence $(r_k)_k$ is nondecreasing and that $r_k > L$ holds for all sufficiently large k. Then $\|(x_{k+1}, y_{k+1}) - (x_k, y_k)\| \longrightarrow 0$.*

Proof.

- We use the approach and the notation of T. Nguyen, Strodiot, and V. Nguyen (2009). Set $\epsilon_k := 1/r_k$ and $h(x) := \|x\|^2$ and note that h is Lipschitz on the compact sets $\tilde{C}_F, \tilde{C}_G$ and that we have $\|x - \hat{x}\|^2 = h(x) - h(\hat{x}) - \langle \nabla h(\hat{x}), x - \hat{x} \rangle$. We can then see that the modified fixed-point iteration above is equivalent to the General Algorithm from T. Nguyen, Strodiot, and V. Nguyen (2009), where in each step we use $\tilde{\theta}_F + \tilde{\theta}_G$ instead of any μ-approximation of $\tilde{\theta}_F + \tilde{\theta}_G$. We then follow the proof of Theorem 1 of T. Nguyen, Strodiot, and V. Nguyen (2009) and note that the argument does not rely on the convexity of the function $\tilde{f}_k$ as long as in each step a global minimum of the auxiliary problem is found. We can thus invoke Theorem 1 of T. Nguyen, Strodiot, and V. Nguyen (2009) in our setting.

- A straightforward computation shows that the bifunction f is pseudomonotone, i. e., if $f(a,b) \geq 0$ then $f(b,a) \leq 0$. Accordingly, we can invoke Proposition 1 from T. Nguyen, Strodiot, and V. Nguyen (2009). Noting that h is strongly convex with modulus $\beta = 2$, we can now use Theorem 2 of T. Nguyen, Strodiot, and V. Nguyen (2009) with $\mu = 1$ and $g(x,y) = 0$ to arrive at $\|(x_{k+1}, y_{k+1}) - (x_k, y_k)\| \longrightarrow 0$.

□

Corollary *Consider the fixed point iteration with (22)–(23) replaced by (26)–(27). Let $(r_k)_k$ be the sequence of regularization parameters used in this fixed-point iteration, and let $(x_k, y_k)_k$ be the sequence generated by this iteration. Suppose that the sequence $(r_k)_k$ is nondecreasing and that $r_k > L$ holds for all sufficiently large k. Then, $(x_k, y_k)_k$ converges to a Nash equilibrium of the modified game with objectives $\tilde{\theta}_F$, $\tilde{\theta}_G$ and feasible sets $\tilde{C}_F$ and $\tilde{C}_G$.*

3.2 A Distributed Algorithm for Solving the Subproblems

Let us now discuss possible ways to solve (24) and (25) and any of their counterparts in a realistic setting. For $i \in F$, we denote by x_i the vector of decision variables controlled by agent i, that is $p_{e,t}$ ($e = (i,j) \in E(F)$, $t = 1, \ldots, T$), $p^J_{i,e,t}$ ($e \in E(G)$, $t = 1, \ldots, T$), and $x_{i,t}$ ($t = 1, \ldots, T$). We can then write $\theta_F(x) = \theta_F(x_1, \ldots, x_m)$, where $m := |F|$. In the given setting, it appears natural to think of each agent i acting semi-independently from all other agents $j \in F$, $j \neq i$. Accordingly, agent i will first try to optimize over the decision variables it controls, i. e., for given $\tilde{x}_j$ ($j \in F, j \neq i$) the agent will solve

$$\begin{aligned} \min \quad & \theta_F(\tilde{x}_1, \ldots, \tilde{x}_{i-1}, x_i, \tilde{x}_{i+1}, \ldots, \tilde{x}_m) + r\|x_i - \hat{x}_i\|^2 && (28) \\ \text{s.t.} \quad & (\tilde{x}_1, \ldots, \tilde{x}_{i-1}, x_i, \tilde{x}_{i+1}, \ldots, \tilde{x}_m) \in C_F(y). && (29) \end{aligned}$$

Such a localized approach has one further advantage: the nonconvexity of the problem is now addressed at a local level with far fewer decision variables involved (i. e., only those associated with agent i). As such, there is the potential to find the global optimum of (28)–(29) in a reasonable amount of time, provided T is not too large and $E(F)$ is not too dense.

To synchronize the optimization efforts of the individual agents, we can consider a simple block-descent algorithm as follows:

1. Choose $(x_1^{(1)}, \ldots, x_m^{(1)}) \in C_F(y)$.
2. $k := 1$
3. For $i = 1, \ldots, m$:
   (a) Set $\tilde{x}_j := x_j^{(k+1)}$, $j = 1, \ldots, i-1$.
   (b) Set $\tilde{x}_j := x_j^{(k)}$, $j = i+1, \ldots, m$.
   (c) Solve (28)–(29) with result $x_i^{(k+1)}$.
   (d) Communicate $x_i^{(k+1)}$ to all agents $j \in F$, $j \neq i$.
4. $k := k + 1$
5. Goto step 3.

Note that Step 3(d) is essentially a broadcasting step, which might not be an appropriate strategy in an adversarial environment. As such, we see the contribution of this subsection as a more theoretical one, and as a first step towards further research. For a more involved scheme with asynchronous communication, we refer the reader to (Luigi Bobbio (2024)).

Convergence results for block-descent algorithms for nonconvex functions are discussed in Attouch et al. (2010), Grippo and Sciandrone (2000), and Yangyang Xu and W. Yin (2013), under quite general conditions. Here, we have the following result.

Theorem. *Denote by $(x^{(k)})_k$ the sequence generated by the above block-descent algorithm. Let the sequence of regularization parameters $(r_k)_k$ be such that $0 < r_L \leq r_k \leq r_U$ for some constants $r_L, r_U > 0$.*

1. *Suppose that $(x^{(k)})_k$ has an accumulation point. Then every accumulation point of $(x^{(k)})_k$ is a critical point to problem (24).*
2. *Denote by Ξ the set of critical points of problem (24). Then $dist(x^{(k)}, \Xi) \to 0$. If Ξ contains uniformly isolated points, then $(x^{(k)})_k$ converges to a point in Ξ.*
3. *Let α be an even integer and let $x^{(1)}$ be sufficiently close to a global minimizer. Then $(x^{(k)})_k$ converges to a global minimizer.*

Proof.

1. This follows directly from Proposition 7 in Grippo and Sciandrone (2000) or from Theorem 2.3 of Yangyang Xu and W. Yin (2013).
2. The sequence $(x^{(k)})_k$ is bounded, as it contains only feasible points. The result then follows from Corollary 2.4 in Yangyang Xu and W. Yin (2013).
3. Let us rewrite problem (24) as

$$\min_{z=(x,y)} f(z) + r_1(x) + r_2(y), \tag{30}$$

where f is the original objective and r_i is the indicator function of $C_F(y)$. The set $C_F(y)$ is semi-algebraic. Likewise, it is easy to see that the function f is semi-algebraic for even integers α. Therefore, the unconstrained problem (30) has a semi-algebraic objective which thus fulfils the Kurdyka-Łojasiewicz inequality at every global solution. Next, we note that f is Lipschitz-continuous on $C_F(y)$, and that the proof of Lemma 2.6 of Yangyang Xu and W. Yin (2013) only requires this Lipschitz-continuity on the set of feasible points. Thus, all assumptions for a slightly strengthened version of Lemma 2.6 of Yangyang Xu and W. Yin (2013) are fulfilled, and the result follows with Corollary 2.7 of Yangyang Xu and W. Yin (2013).

□

3.3 A Model Predictive Control Framework to Handle Dynamic Environments

To handle uncertain information in the environment and, in particular, dynamically changing parameters of the underlying problem, we propose to use a *rolling horizon* or *model predictive control* framework. By a rolling horizon approach, the acting agents or UAVs can react to changes in the environment as soon as sensors on the UAVs detect corresponding changes or are informed of such a change by other members of their fleet. This approach is widely used in engineering due to its simplicity and versatility, see e. g., Findeisen, Allgöwer, and Biegler (2007) and Rawlings, Mayne, Diehl, et al. (2017).

The rest of this section follows closely from previous work by Atayev, Fliege, and Zemkoho (2024). Let us consider the time interval $[0, T]$ over which we optimise. As *scheduling horizon* we chose $[0, T]$. For $T_0 = 0 < T_1 < T_2 \cdots$ we define *prediction horizons* $[T_i, T]$ and *control horizons* $[T_i, T_{i+1}]$, $i = 0, 1, \ldots$. At the ith step, beginning with $i = 0$, we compute a solution $(x^{(i)}, y^{(i)})$ to the Nash game under consideration, where the game is played over the ith prediction horizon, $[T_i, T]$, and we have updated the constraints of the game by taking into account all the actions of all agents at times $t < T_i$. For example, suppose that at some time step $t < T_i$ agent j expends energy $p_{e,t} > 0$ to communicate with agent k along the edge $e = (j, k)$. We then update the right-hand side of the power constraint of agent j by $P_j^{max} := P_j^{max} - p_{e,t}$, with a similar update if power is also used for jamming or moving. Within the ith control horizon, $[T_i, C_i]$, the agents then enact all decisions encoded in $(x^{(i)}, y^{(i)})$. The agents will update their current estimate of all uncertainties of their environment whilst enacting these decisions. Once the agents have reached the beginning of the next prediction horizon, that is, the end of their current control horizon, they can make use of the updated information when solving the Nash game again over the next prediction horizon, leading to $(x^{(i+1)}, y^{(i+1)})$. Repeat this process until agents have enacted decisions over the full interval $[0, T]$. The overall control loop, to be executed for times $t = 1, \ldots, T$ can thus be written as follows. We write $x = (x_1, \ldots, x_T)$ and $y = (y_1, \ldots, y_T)$, where x_t, y_t denote the decisions made at time t. Moreover, we highlight the influence of dynamically changing parameters by writing the payoff functions as $\theta_F(x, y, \mathbf{p}_t)$ and $\theta_F(x, y, \mathbf{p}_t)$ instead of $\theta_F(x, y), \theta_G(x, y)$ where $\mathbf{p}_t$ denotes all parameters of the environment observed at time t and before.

1. We have given previous decisions $\tilde{\mathbf{x}}_{t-1} := (\tilde{x}_1, \ldots, \tilde{x}_{t-1})$ and $\tilde{\mathbf{y}}_{t-1} := (\tilde{y}_1, \ldots, \tilde{y}_{t-1})$ and previous parameters $\mathbf{p}_{t-1} := (p_1, \ldots, p_{t-1})$.
2. Observe the parameter vector p_t and set $\mathbf{p}_t := (p_1, \ldots, p_t)$.
3. Update the sets of feasible points, based on $\tilde{\mathbf{x}}_{t-1}, \tilde{\mathbf{y}}_{t-1}$, and $\mathbf{p}_t$.
4. Solve the Nash game with strategies $(x_t, \ldots, x_T)$ and $(y_t, \ldots, y_T)$ and objectives $\theta_F(\tilde{\mathbf{x}}_{t-1}, \bullet; \tilde{\mathbf{y}}_{t-1}, \bullet; \mathbf{p}_t)$ and $\theta_G(\tilde{\mathbf{x}}_{t-1}, \bullet; \tilde{\mathbf{y}}_{t-1}, \bullet; \mathbf{p}_t)$.
5. Denote the computed equilibrium solution by $(x_t, \ldots, x_T)$ and $(y_t, \ldots, y_T)$.
6. Set $\tilde{x}_t := x_t$ and $\tilde{y}_t := x_t$.

We refer to Atayev, Fliege, and Zemkoho (2024) for further details, particularly concerning the choice of the lengths of the control horizons and the possibility of adding smoothness constraints to the overall trajectory constructed.

4 Numerical Experiments

For our numerical experiments, we used the AMPL modelling language v. 20240606 to implement various instances of the Nash game described here. We used KNITRO v. 14.0.0 (Byrd, Nocedal, and Waltz, 2006) to solve the NLP problems (26)–(27) occurring in each step of our algorithm. The following settings were applied: presolve and absolute feasibility tolerances equal to 10^{-6}; absolute optimality tolerance 10^{-6}; multistart enabled. We allow a maximum of 30 fixed-point iterations for the solution of each instance. The stopping criteria for the fixed-point algorithm was set to $\varepsilon := \|(x_{k+1}, y_{k+1}) - (x_k, y_k)\| \leq 10^{-3}$. We use $r = 10^{-2}$ as the starting value for the regularization parameter r. At each fixed-point iteration, r is multiplied by 2. Our experiments were executed on an AMD Ryzen 3 5400U laptop with 2.60GHz and 16GB of RAM running under Linux Mint 21.0 64bits.

We consider two fleets, F ('the Latins') and G ('the Greeks'), with varying numbers of agents. We use Latin indices for members of fleet F and Greek indices for members of fleet G. In some of our experiments, we set the maximal velocity of some agents to zero, indicating that this agent is a ground asset.

Table 1 shows the values of the constants of the model defined in Section 2. We note that

Parameter	Value
ϱ	1.0
α	2.0
ϵ	0.2
σ	1.0

Table 1: Parameters defining the system model.

$\alpha = 2$ is a typical choice for simulations and that choosing $\varrho = 1$ is equivalent to scaling through all numerators and denominators in the objectives by ϱ and replacing σ^2 by σ^2/ϱ.

4.1 Illustrative Examples

In what follows we detail each numerical example together with their respective choice of remaining parameters.

Example 1 In this example, fleets $F = \{a, b\}$ and $G = \{\alpha, \beta\}$ start their missions at time $t = 1$ and finish at time $T = 5$. Table 2 shows the parameters for the different agents used in this example, where x_0 and y_0 refer to the starting positions of the UAVs in the (x, y)-plane. The

Agent	x_0	y_0	P^{max}	c	δ
a	1.0	1.0	1.0	0.2	0.5
b	-1.0	1.0	1.0	0.2	0.5
α	1.0	-1.0	1.0	0.2	0.5
β	-1.0	-1.0	1.0	0.2	0.5

Table 2: Choice of UAV parameters for Example 1.

communication network in this example is represented by the edge sets $E(F) = \{(a, b), (b, a)\}$ and $E(G) = \{(\alpha, \beta), (\beta, \alpha)\}$, i. e., each agent in fleets F and G communicates with the other agent in their fleet, respectively. Further, we use $J(e) = G$ for all $e \in E(F)$ and $J(e) = F$ for all $e \in E(G)$, meaning that agents in fleets F and G are capable of jamming communication along all arcs in $E(G)$ and $E(F)$, respectively. Finally, we set $\omega_F = \omega_G = 10^{-2}$, meaning that both fleets are mostly interested in jamming the communications of the other fleet.

Figure 2 shows the optimal trajectories for both fleets as computed by our fixed-point algorithm, while Figure 3 shows the power expenditure, both for communication and for jamming, across both fleets for all time steps. As it can be seen, the strategies deployed by the two fleets are nonsymmetric: both fleets fly their agents close to the agents of the opposite team to facilitate jamming, but only agent a uses a non-negligible amount of its battery power for communication. Finally, Figure 4 depicts the objective function values observed over the steps of our fixed-point algorithm. As it can be seen we arrive fairly quickly at $\theta_F = \theta_G$, indicating that we have achieved convergence.

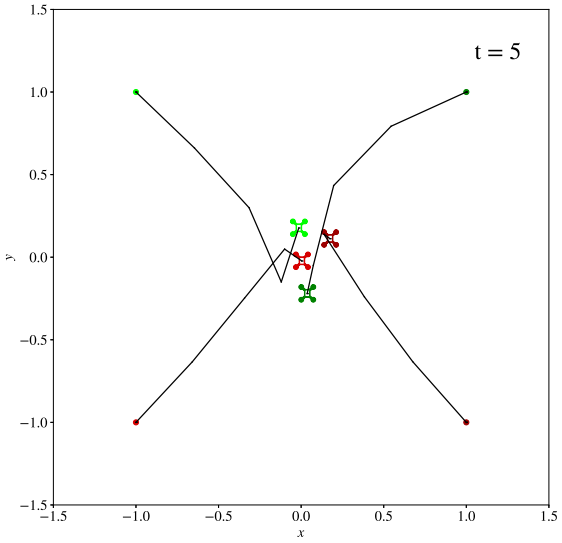


Figure 2: Optimal trajectories as computed by our fixed-point algorithm for Example 1. All UAVs reduce the distance to members of the opposite fleet as much as possible to increase their jamming power.

Example 2 We modify Example 1 by considering nonsymmetric starting positions, see Table 3. We use the same communication graphs and jamming capabilities as before.

As shown in Figures 5 and 6, in this example members of fleet F spend their power exclusively for communication and jamming but do not move in space. In contrast, fleet G tries to jam the communications of fleet F by shortening the distance to fleet F whilst simultaneously also facilitating some communication between its members. Figure 7 shows the objective function values over the course of our fixed-point algorithm, again indicating rapid convergence.

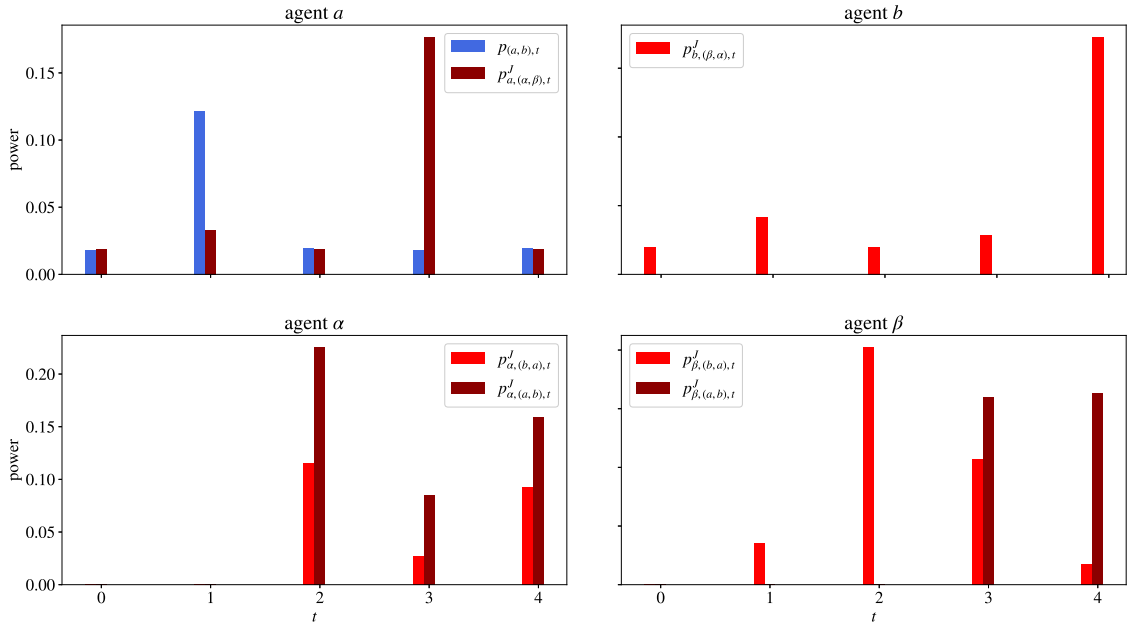


Figure 3: Optimal power expenditure over time as computed by our fixed-point algorithm for Example 1. Blue bars represent power used for communicating, and red and brown bars represent power used for jamming. Power expenditures smaller than 10^{-2} are not included in the histogram. As it can be seen, only the UAV a uses a non-negligible amount of power for communication.

Agent	x_0	y_0	P^{max}	c	δ
a	1.3	2.7	1.0	0.2	0.5
b	-1.1	1.5	1.0	0.2	0.5
α	0.7	-0.9	1.0	0.2	0.5
β	-2.1	-2.2	1.0	0.2	0.5

Table 3: UAV parameters for Example 2.

Example 3 In this example, fleets $F = \{a, b, g\}$ and $G = \{\alpha, \beta, \gamma\}$ start at time $t = 1$ and finalize their mission at time $T = 5$. Agents g and γ represent ground units which do not move during the planning horizon. Table 4 provides the UAV parameters for this example. As edge sets we use $E(F) = \{(a, b), (a, g), (b, a), (b, g), (g, a), (g, b)\}$ and $E(G) = \{(\alpha, \beta), (\alpha, \gamma), (\beta, \alpha), (\beta, \gamma), (\gamma, \alpha), (\gamma, \beta)\}$. We allow all UAVs to jam all communication edges, but do not provide ground units with jamming capabilities.

Figures 8 and 9 depict the computed solution. Agent a and agent α move as closely together as possible (roughly at coordinates (-1, 0.3)). At that time, agent a jams the communication between α and the ground unit λ. Agent a then flies back to its starting position, while agent α follows a and jams the communication between a and ground unit g. Figure 10 shows the objective function values computed at each step of our fixed-point algorithm.

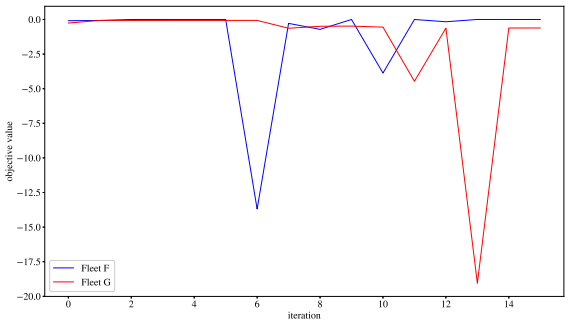


Figure 4: Objective function values per fixed-point iteration for each fleet in Example 1.

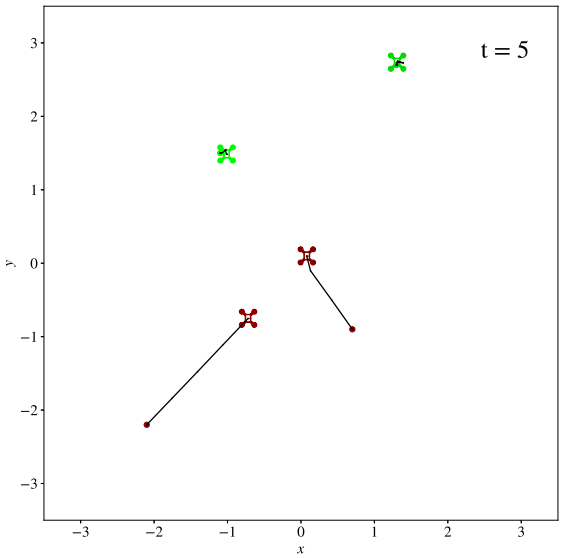


Figure 5: Optimal trajectories as computed by our fixed-point algorithm for Example 2. The two members of fleet F (green) do not move. Both members of fleet G (red) reduce their distance to the opposing fleet to increase their jamming capabilities, while simultaneously also shortening the distance between themselves.

Example 4 This example highlights the effectiveness of our MPC approach in a dynamically changing environment. Here we consider fleets with different numbers of agents. We start with having $F = \{a, b, c, g\}$ and $G = \{\alpha, \beta, \gamma\}$ starting their mission at time $t = 1$, while the mission ends at time $T = 10$. There is also an additional UAV, $c \in F$, which is only activated at step $t = 5$. As in the previous example, agents g γ represent ground units. Table 5 shows the UAV parameters for this example. The communication network in this example is represented by the sets $E(F) = \{(a, b), (a, c), (b, a), (b, c), (c, a), (c, b), (c, g), (g, c)\}$. As one can see, agents a, b and c can communicate among themselves, but only agent c can communicate with the ground unit g.

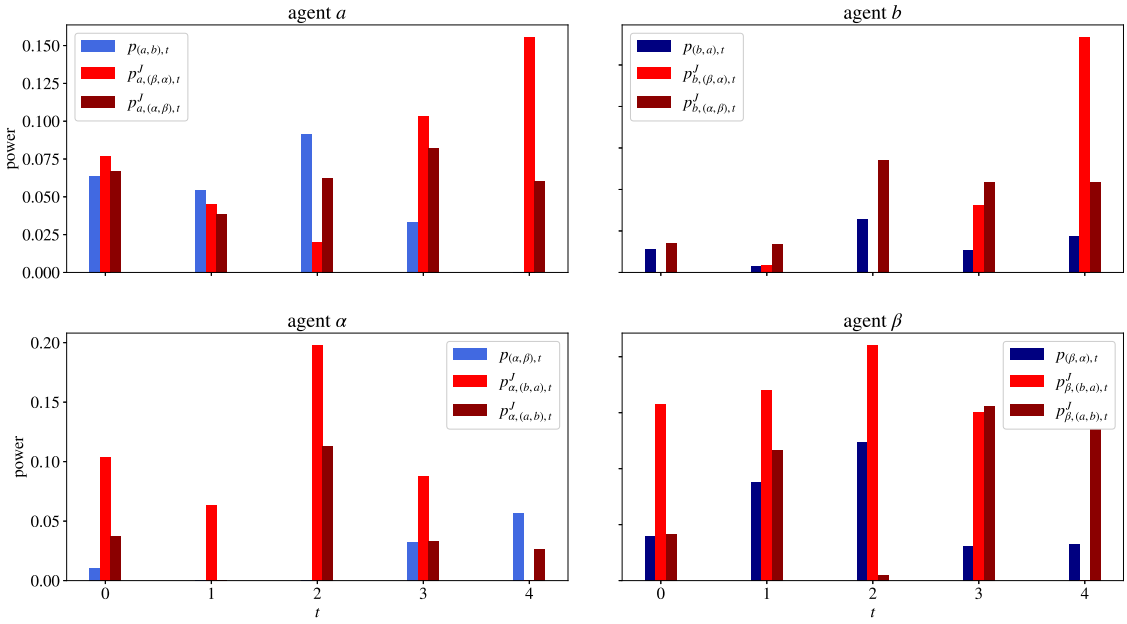


Figure 6: Optimal power expenditure over time as computed by our block-descent algorithm for Example 2.

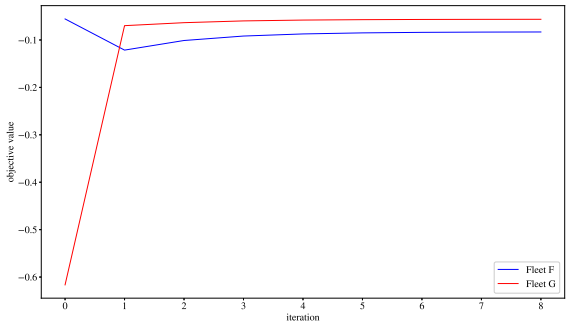


Figure 7: Objective function values per fixed-point iteration for each fleet/player in Example 2.

Thus, agent c can be seen as a relay node between agents a, b and g. In the communication graph for fleet G, agents α and β can communicate among themselves and with the ground unit γ, i.e., $E(G) = \{(\alpha,\beta),(\alpha,\gamma),(\beta,\alpha),(\beta,\gamma),(\gamma,\alpha),(\gamma,\beta)\}$. The jamming capabilities are described by the index sets $J(e) = \{\alpha,\beta\}$ for all $e \in E(F) \setminus I(g)$ and $J(e) = \{a,b,c\}$ for all $e \in E(G) \setminus I(\gamma)$, where $I(g)$ and $I(\gamma)$ are the edges incident to g and γ, respectively.

Figures 11 and 12 depict the optimal solution computed at time $t = 1$ for the time frame $t = 1,\ldots,10$, without knowledge that a new UAV will appear at time step 5. Fleet F does not move at all and uses all its power budget for communication and jamming. Fleet G moves closer

Agent	x_0	y_0	P^{max}	c	δ
a	1.0	1.0	1.0	0.2	0.5
b	-1.0	1.0	1.0	0.2	0.5
g	0.0	1.0	1.0	1.0	0.0
α	1.0	-1.0	1.0	0.2	0.5
β	-1.0	-1.0	1.0	0.2	0.5
γ	0.0	-1.0	1.0	1.0	0.0

Table 4: UAV parameters for Example 3.

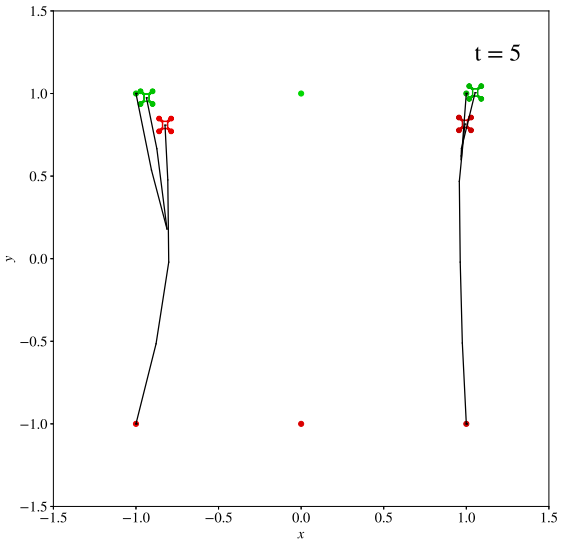


Figure 8: Optimal trajectories as computed by our fixed-point algorithm for Example 3.

Agent	x_0	y_0	P^{max}	c	δ
a	1.0	1.0	1.0	0.2	0.5
b	-1.0	1.0	1.0	0.2	0.5
c	-1.5	0.5	1.0	0.2	0.5
g	0.0	2.0	1.0	1.0	0.0
α	1.0	-1.0	1.0	0.2	0.5
β	-1.0	-1.0	1.0	0.2	0.5
γ	0.0	-2.0	1.0	1.0	0.0

Table 5: UAV parameters for Example 4.

to fleet F to facilitate jamming, and the two UAVs also move closer together to improve their communications.

We now consider the situation at time $t = 5$, after which both fleets have executed their optimal strategies as depicted in Figures 11 and 12 for the time steps $t = 1, 2, 3, 4$. The UAV c in fleet F

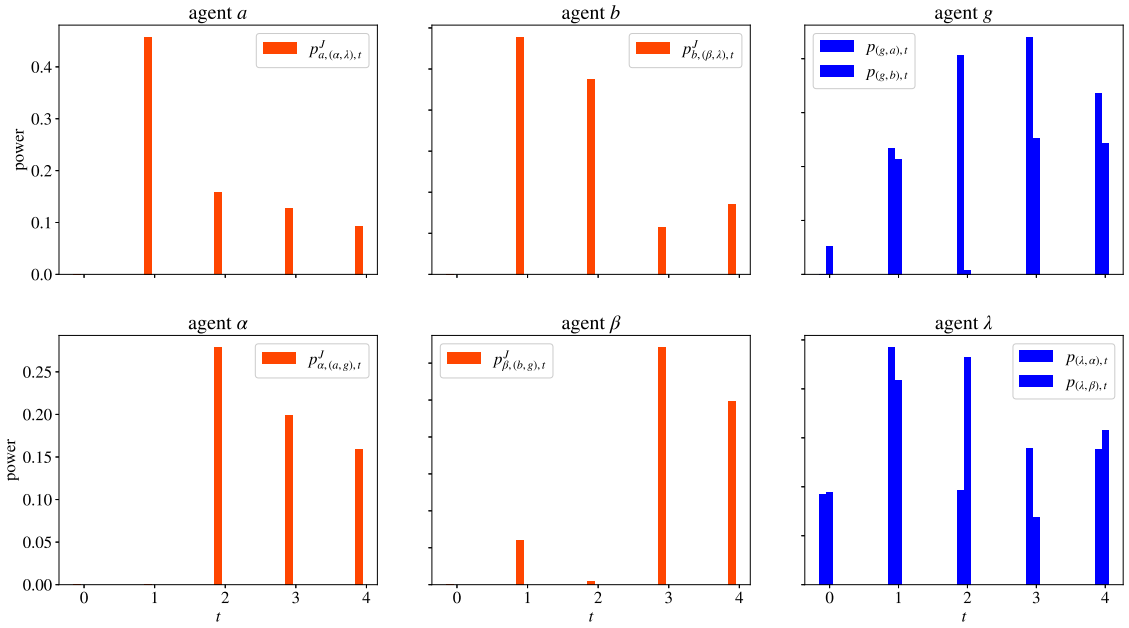


Figure 9: Power allocation over time as computed by our fixed-point algorithm for Example 3. Power values smaller than 10^{-2} are not depicted.

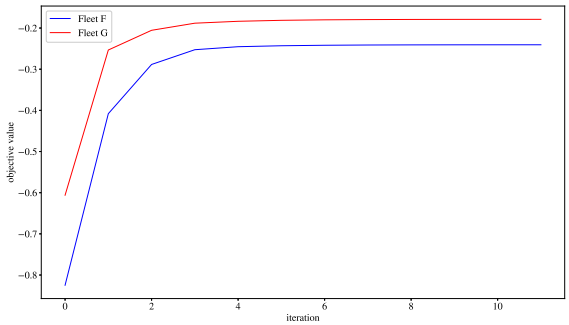


Figure 10: Objective function values per fixed-point iteration for each fleet/player in Example 3.

is now activated and its presence is made known to both fleets. Both players now reoptimize the remaining Nash game over time steps $t = 5, \ldots, 10$. Figures 13 and 14 show the solutions found. As one can see, the strategy of fleet F has changed markedly and can be described as having a much more aggressive stance than before.

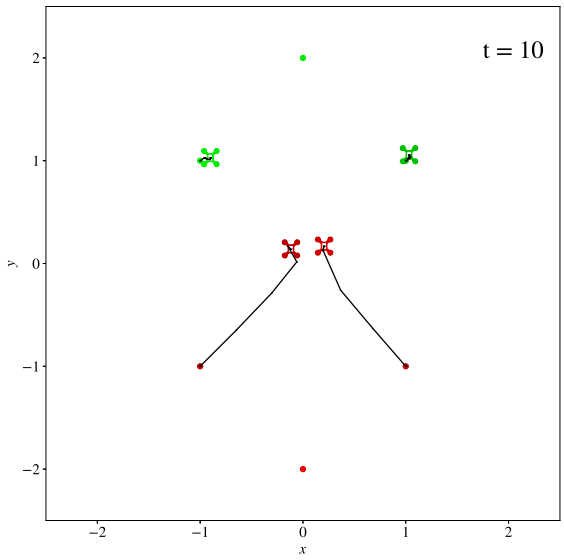


Figure 11: Optimal trajectories as computed by our fixed-point algorithm for Example 4 at time $t = 1$ for the full mission length, without knowledge of the dynamics of the setting.

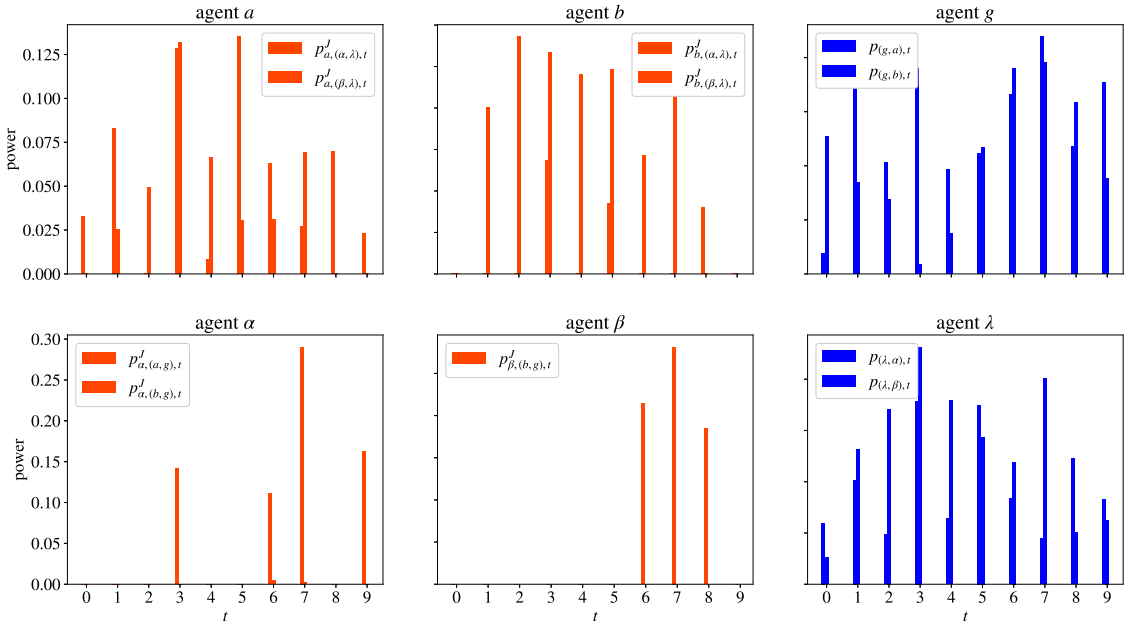


Figure 12: Optimal power expenditure over time as computed by our fixed-point algorithm for Example 4, at time $t = 1$ for the full mission length, without knowledge of the dynamics of the setting.

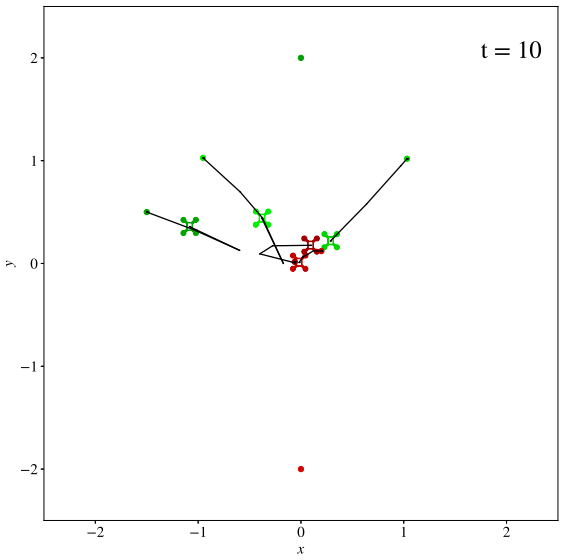


Figure 13: Optimal trajectories as computed by our fixed-point algorithm for Example 4 at time $t = 5$ for the remaining mission length.

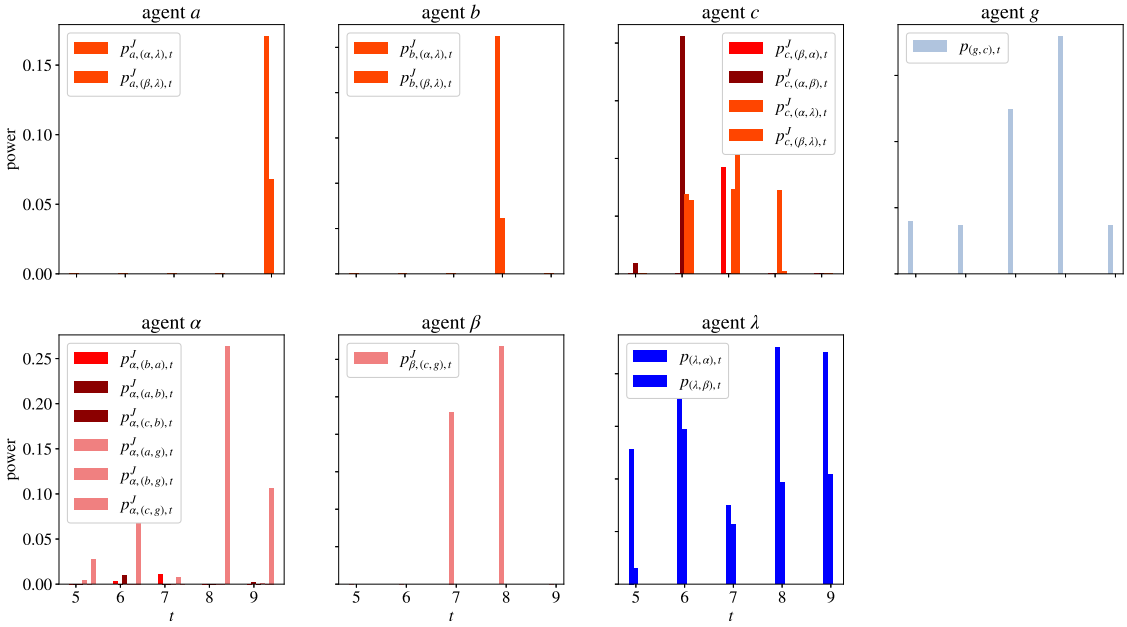


Figure 14: Optimal power expenditure over time as computed by our fixed-point algorithm for Example 4, at time $t = 5$ for the remaining mission length.

4.2 Computational Performance

Here we investigate the computational performance of the proposed approach across different problem instances, where we increase the problem parameter T, i. e., the number of time steps available for the mission. Table 7 summarizes our results. The column headers of the table denote the following:

T	number of time steps in the problem,
it^{fp}	number of fixed-point iterations,
Avg. T^{fp}	average time per fixed-point iteration (in seconds),
Avg. it^{nlp}	average number of NLP iterations (per NLP call) per fixed-point iteration,
Avg. T^{nlp}	average time per NLP call (in seconds),
$T(s)$	total time for the fixed-point algorithm (in seconds).

Table 7 shows how our algorithm scales as T increases. For these experiments, we consider a slight variant of Example 1, in which we keep the same number of agents per fleet and the same communication and jamming capabilities, but use slightly different parameters for the UAVs. These are presented in Table 6. Note that the NLPs solved in each step of the fixed-point algorithm increase in size with N; they have approximately $18T$ variables and $6T$ nonlinear constraints.

Agent	x_0	y_0	P^{max}	c	δ
a	1.0	1.0	1.0	0.6	0.05
b	-1.0	1.0	1.0	0.5	0.1
α	1.0	-1.0	1.0	0.6	0.05
β	-1.0	-1.0	1.0	0.5	0.1

Table 6: Choice of UAV parameters for the modified version of Example 1, as considered in Subsection 4.2.

One can observe that the total computation time increases significantly for $T \geq 50$. The number of fixed-point iterations remains within a reasonable range across all instances, never reaching the maximum of 30, thus showing that the method converges consistently. However, the average time per fixed-point iteration grows considerably for larger values of T, indicating that while the convergence properties remain stable, the cost of each iteration increases as the problem size expands, as expected. The data further shows an increase in the average number of NLP iterations per NLP call, and their corresponding time, especially for $T = 100$. This suggests that while the method is efficient for smaller planning horizons, future work may need to focus on optimizing the time per iteration or reducing the number of NLP calls to ensure scalability for larger problems.

In summary, the proposed approach demonstrates a satisfactory convergence and effective performance across all problem instances. The average time per fixed-point iteration and the total NLP time are the primary factors contributing to the rise in computational time, particularly for larger values of T. On the other hand, we highlight that even the largest computation time needed in our experiments (c.a. 200s) can considered reasonable in practical applications.

T	it^{fp}	Avg. T^{fp}(s)	Avg. it^{nlp}	Avg. T^{nlp}(s)	T(s)
5	9	0.02	49.22	0.02	0.20
10	8	0.02	90.75	0.02	0.19
20	8	0.25	225.75	0.08	2.00
30	7	0.29	292.86	0.15	2.00
40	10	0.60	350.50	0.26	6.00
50	17	1.41	721.88	0.71	24.00
100	15	14.47	2865.80	7.23	217.00

Table 7: Results for the scenario from Example 1 with varying T.

5 Conclusion

In this paper, we introduced a novel game-theoretic model for jamming and communication between fleets of unmanned aerial vehicles (UAVs). The problem was framed as a Nash game, where each fleet aims to optimize its communication while simultaneously jamming the communication of the opposing fleet. By utilizing properties of the electromagnetic spectrum and the mobility of agents, our model captures both the complexity of jamming strategies, the dynamics of inter-agent communication, and important spatial characteristics of the problem. We proposed a fixed-point algorithm to solve the problem, providing convergence guarantees under mild conditions.

Our computational experiments demonstrated the effectiveness of the approach across various problem sizes. The proposed algorithm consistently converged in all cases, indicating the efficacy of the method even for larger instances.

The results highlight the potential of the proposed approach for real-world applications in electronic warfare and UAV fleet management, particularly in scenarios where both communication and jamming are key strategic objectives. Future work could focus on improving the scalability of the method by speeding up the solution of the NLP subproblems, as well as exploring parallelization techniques for distributed settings. Additionally, extending the model to include more complex dynamics, such as stochastic elements to manage uncertainty or multi-fleet interactions would further enhance its practical relevance.

In conclusion, the proposed method offers a flexible and effective framework for addressing jamming games for fleets of UAVs, providing further insights into the interplay between communication and interference in dynamic adversarial environments.

6 Statements and Declarations

Funding. The authors thank UFPE/PPGEP-CAA for supporting this project through the internal call PROPESQI-PROPG-DRI Prof. Visitante Nº. 13/2022 and CAPES through project 88881.917250/2023-01.

Availability of data and materials. Datasets abd code can be accessed by contacting the first author, Walton P. Coutinho.

Acknowledgements. We are indebted to colleagues from the Defence, Science and Technology Laboratory (DSTL) of the United Kingdom who provided valuable comments on a preliminary version of this manuscript.

References

Atayev, Anvarbek, Jörg Fliege, and Alain Zemkoho (2024). "Trajectory optimization of unmanned aerial vehicles in the electromagnetic environment". In: *Optimization and Engineering*, pp. 1–40.

Attouch, Hédy et al. (2010). "Proximal alternating minimization and projection methods for nonconvex problems: An approach based on the Kurdyka-Łojasiewicz inequality". In: *Mathematics of operations research* 35.2, pp. 438–457.

Bhattacharya, Sourabh and Tamer Başar (2010). "Game-theoretic analysis of an aerial jamming attack on a UAV communication network". In: *Proceedings of the 2010 American Control Conference*, pp. 818–823. URL: https://api.semanticscholar.org/CorpusID:7276994.

— (2011a). "Spatial approaches to broadband jamming in heterogeneous mobile networks: a game-theoretic approach". In: *Autonomous Robots* 31.4, pp. 367–381.

— (2011b). "Spatial approaches to broadband jamming in heterogeneous mobile networks: a game-theoretic approach". In: *Autonomous Robots* 31, pp. 367–381. URL: https://api.semanticscholar.org/CorpusID:30214890.

— (2012). "Multi-layer hierarchical approach to double sided jamming games among teams of mobile agents". In: *2012 IEEE 51st IEEE Conference on Decision and Control (CDC)*, pp. 5774–5779. URL: https://api.semanticscholar.org/CorpusID:27083485.

— (2013). "Differential Game-Theoretic Approach to a Spatial Jamming Problem". In: *Advances in Dynamic Games: Theory, Applications, and Numerical Methods for Differential and Stochastic Games*. Ed. by Pierre Cardaliaguet and Ross Cressman. Boston, MA: Birkhäuser Boston, pp. 245–268. ISBN: 978-0-8176-8355-9. DOI: 10.1007/978-0-8176-8355-9_13. URL: https://doi.org/10.1007/978-0-8176-8355-9_13.

Bhattacharya, Sourabh, Abhishek K. Gupta, and Tamer Başar (2013). "Jamming in mobile networks: A game-theoretic approach". In: *Numerical Algebra, Control and Optimization* 3, pp. 1–30. URL: https://api.semanticscholar.org/CorpusID:58726904.

Bhattacharya, Sourabh, Ali Khanafer, and Tamer Başar (2016). "A Double-Sided Jamming Game with Resource Constraints". In: *Advances in Dynamic and Evolutionary Games: Theory, Applications, and Numerical Methods*. Ed. by Frank Thuijsman and Florian Wagener. Cham: Springer International Publishing, pp. 209–227. ISBN: 978-3-319-28014-1. DOI: 10.1007/978-3-319-28014-1_10. URL: https://doi.org/10.1007/978-3-319-28014-1_10.

Byrd, Richard H, Jorge Nocedal, and Richard A Waltz (2006). "K nitro: An integrated package for nonlinear optimization". In: *Large-scale nonlinear optimization*, pp. 35–59.

Feng, Zikai et al. (2023). "Approximating Nash equilibrium for anti-UAV jamming Markov game using a novel event-triggered multi-agent reinforcement learning". In: *Neural Networks* 161, pp. 330–342.

Findeisen, Rolf, Frank Allgöwer, and Lorenz T Biegler (2007). *Assessment and future directions of nonlinear model predictive control*. Vol. 358. 7. Springer.

Flåm, Sjur Didrik and Anatoly S Antipin (1996). "Equilibrium programming using proximal-like algorithms". In: *Mathematical Programming* 78.1, pp. 29–41.

Fu, Shu et al. (2023). "Joint power allocation and 3D deployment for UAV-BSs: A game theory based deep reinforcement learning approach". In: *IEEE Transactions on Wireless Communications* 23.1, pp. 736–748.

Grippo, Luigi and Marco Sciandrone (2000). "On the convergence of the block nonlinear Gauss–Seidel method under convex constraints". In: *Operations research letters* 26.3, pp. 127–136.

Gupta, Abhishek et al. (2012). "A dynamic transmitter-jammer game with asymmetric information". In: *2012 IEEE 51st IEEE Conference on Decision and Control (CDC)*. IEEE, pp. 6477–6482.

Khanafer, Ali, Sourabh Bhattacharya, Tamer Bas, et al. (2011). "Adaptive resource allocation in jamming teams using game theory". In: *2011 International Symposium of Modeling and Optimization of Mobile, Ad Hoc, and Wireless Networks*. IEEE, pp. 395–400.

Khanafer, Ali, Sourabh Bhattacharya, and Tamer Başar (2011). "Adaptive resource allocation in jamming teams using game theory". In: *2011 International Symposium of Modeling and Optimization of Mobile, Ad Hoc, and Wireless Networks*, pp. 395–400. URL: https://api.semanticscholar.org/CorpusID:1193353.

Li, Yongcheng et al. (2024). "Intelligent Jamming Strategy for Wireless Communications Based on Game Theory". In: *IEEE Access*.

Liu, Yuan et al. (2024). "A Game Theoretical Anti-jamming Beamforming Approach for Integrated Sensing and Communications Systems". In: *IEEE Transactions on Vehicular Technology*.

Lu, Xiaozhen et al. (2017). "Anti-jamming communication game for UAV-aided VANETs". In: *GLOBECOM 2017-2017 IEEE Global Communications Conference*. IEEE, pp. 1–6.

Luigi Bobbio Jörg Fliege, Antonio Martinez-Sykora (2024). *Distributed Task Assignment in a Swarm of UAVs*. Submitted. Available at https://optimization-online.org/2024/02/distributed-task-assignment-in-a-swarm-of-uavs/. University of Southampton.

Lv, Shichao et al. (2017). "Anti-jamming power control game in unmanned aerial vehicle networks". In: *GLOBECOM 2017-2017 IEEE Global Communications Conference*. IEEE, pp. 1–6.

Mkiramweni, Mbazingwa Elirehema et al. (2019). "A survey of game theory in unmanned aerial vehicles communications". In: *IEEE Communications Surveys & Tutorials* 21.4, pp. 3386–3416.

Nguyen, TTV, Jean-Jacques Strodiot, and VH Nguyen (2009). "A bundle method for solving equilibrium problems". In: *Mathematical programming* 116, pp. 529–552.

Palomar, Daniel Pérez, Mats Bengtsson, and Björn Ottersten (2005). "Minimum BER linear transceivers for MIMO channels via primal decomposition". In: *IEEE Transactions on Signal Processing* 53.8, pp. 2866–2882.

Pang, Jong-Shi and Gesualdo Scutari (2011). "Nonconvex games with side constraints". In: *SIAM Journal on Optimization* 21.4, pp. 1491–1522.

Parras, Juan, Jorge del Val, et al. (2016). "A new approach for solving anti-jamming games in stochastic scenarios as pursuit-evasion games". In: *2016 IEEE Statistical Signal Processing Workshop (SSP)*, pp. 1–5. URL: https://api.semanticscholar.org/CorpusID:16018797.

Parras, Juan, Santiago Zazo, et al. (2017). "Pursuit-evasion games: a tractable framework for antijamming games in aerial attacks". In: *EURASIP Journal on Wireless Communications and Networking* 2017, pp. 1–15. URL: https://api.semanticscholar.org/CorpusID:29926870.

Rawlings, James Blake, David Q Mayne, Moritz Diehl, et al. (2017). *Model predictive control: theory, computation, and design*. Vol. 2. Nob Hill Publishing Madison, WI.

Rosen, J Ben (1965). "Existence and uniqueness of equilibrium points for concave n-person games". In: *Econometrica: Journal of the Econometric Society*, pp. 520–534.

Su, Yueyue et al. (2024). "Cooperative anti-jamming and interference mitigation for UAV networks: A local altruistic game approach". In: *China Communications* 21.2, pp. 183–196.

Xu, Jiangwei et al. (2021). "Anti-jamming strategy based on game theory in single-channel UAV communication network". In: *2021 Sixth International Conference on Fog and Mobile Edge Computing (FMEC)*. IEEE, pp. 1–7.

Xu, Yangyang and Wotao Yin (2013). "A block coordinate descent method for regularized multi-convex optimization with applications to nonnegative tensor factorization and completion". In: *SIAM Journal on imaging sciences* 6.3, pp. 1758–1789.

Xu, Yifan et al. (2018). "A one-leader multi-follower Bayesian-Stackelberg game for anti-jamming transmission in UAV communication networks". In: *Ieee Access* 6, pp. 21697–21709.

Yin, Ziyan et al. (2024). "UAV Communication Against Intelligent Jamming: A Stackelberg Game Approach With Federated Reinforcement Learning". In: *IEEE Transactions on Green Communications and Networking*.

Zhang, Tianxian et al. (2023). "Task assignment in UAV-enabled front jammer swarm: A coalition formation game approach". In: *IEEE Transactions on Aerospace and Electronic Systems*.