Declarative vs Rule-based Control for Flocking Dynamics

Usama Mehmood
Department of Computer Science,
Stony Brook University, USA

Nicola Paoletti
Department of Computer Science,
Stony Brook University, USA

Dung Phan
Department of Computer Science,
Stony Brook University, USA

Radu Grosu
Cyber-Physical Systems Group,
Technische Universitat Wien, Austria

Shan Lin
Department of Electrical and
Computer Engineering, Stony Brook
University, USA

Scott D. Stoller
Department of Computer Science,
Stony Brook University, USA

Ashish Tiwari
SRI International, USA

Junxing Yang
Department of Computer Science,
Stony Brook University, USA

Scott A. Smolka
Department of Computer Science,
Stony Brook University, USA

ABSTRACT

The popularity of rule-based flocking models, such as Reynolds' classic flocking model, raises the question of whether more declarative flocking models are possible. This question is motivated by the observation that declarative models are generally simpler and easier to design, understand, and analyze than operational models. We introduce a very simple control law for flocking based on a cost function capturing cohesion (agents want to stay together) and separation (agents do not want to get too close). We refer to it as *d*eclarative flocking (DF). We use model-predictive control (MPC) to define controllers for DF in centralized and distributed settings. A thorough performance comparison of our declarative flocking with Reynolds' model, and with more recent flocking models that use MPC with a cost function based on lattice structures, demonstrate that DF-MPC yields the best cohesion and least fragmentation, and maintains a surprisingly good level of geometric regularity while still producing natural flock shapes similar to those produced by Reynolds' model. We also show that DF-MPC has high resilience to sensor noise.

1 INTRODUCTION

Flocking is a collective behavior exhibited by a large number of interacting agents possessing a common group objective [4]. The term is most commonly associated with birds, and more recently, drones. Examples include foraging for food, executing a predator-avoidance maneuver, and engaging in migratory behavior.

With the introduction of Reynolds' model [7, 8], *rule-based control* became the norm in the flocking community. Specifically, in this model, at each time-step, each agent executes a control law given in terms of the weighted sum of three competing forces to determine its next acceleration. Each of these forces has its own rule: *separation* (keep a safe distance away from your neighbors), *cohesion* (move towards the centroid of your neighbors), and *alignment* (steer toward the average heading of your neighbors). As the descriptions suggest, these rules are executed by each agent in a distributed environment with limited-range sensing and no communication.

The popularity of Reynolds' model and its many variants raises the question: Is there a more abstract *declarative* form of control for flocking? This question is important because declarative models are generally simpler and easier to design, understand, and analyze than operational models. This is analogous to declarative programs (e.g., functional programs and logic programs) being easier to write and verify than imperative programs.

We show that the answer to this question is indeed positive by providing a very simple control law for flocking based on a *cost function* comprising two main terms: *cohesion* (the average squared distance between all pairs of agents) and *separation* (a sum of inverse squared distances, except this time between pairs of agents within each other's sensing range). That is it. For example, no term representing velocity alignment is needed. The cost function specifies what we want as the goal, and is hence declarative. In contrast, the update rules in Reynolds' model aim to achieve an implicit goal and hence are operational. Executing declarative control amounts to finding the right balance between attracting and repelling forces between agents. We refer to this approach as *Declarative Flocking* (DF). We use MPC (model-predictive control) to define controllers for DF, and refer to this approach as *DF-MPC*. We define a centralized version of DF-MPC, which requires communication, and a distributed version, which does not.

Previous MPCs for flocking exist, e.g., [11–13]. Most of these MPCs are designed to conform to the α-lattice model of flocking proposed in [4]. α-lattices impose a highly regular structure on flocks: all neighboring agents are distance d apart, for a specified constant d. This kind of structure is seen in some settings, such as beehives, but is not expected in many other natural and engineered settings, and it is not imposed by Reynolds' model.

In this paper, we show, via a thorough performance evaluation, how centralized and distributed DF-MPC compare to Reynolds' rule-based approach [7, 8], Olfati-Saber's potential-based approach [4], a variant of Zhan and Li's centralized lattice-based MPC approach [10, 11], and Zhang *et al.*'s distributed lattice-based MPC approach [12]. We consider performance measures that capture multiple dimensions of flocking behavior: number of sub-flocks (flock fragmentation), maximum sub-flock diameter (cohesion), velocity convergence, and a new parameter-free measure of the geometric regularity of the formation.

Our experimental results demonstrate that DF-MPC yields the best cohesion and least fragmentation, and produces natural flock shapes like those produced by Reynolds' model. Also, distributed

DF-MPC maintains a surprisingly good level of geometric regularity. We also analyze the resiliency of DF-MPC and the lattice-based MPC approaches by considering the impact of sensor noise. Our results demonstrate a remarkably high level of resiliency on the part of DF-MPC in comparison with these other approaches.

2 MODELS OF FLOCKING BEHAVIOR

We consider a set of dynamic *agents* $\mathcal{B} = \{1, \ldots, n\}$ that move according to the following discrete-time equation of motion:

$$x_i(k+1) = x_i(k) + dT \cdot v_i(k),\ v_i(k) \in V \tag{1}$$

$$v_i(k+1) = v_i(k) + dT \cdot a_i(k),\ a_i(k) \in A, \tag{2}$$

where $x_i(k), v_i(k), a_i(k) \in \mathbb{R}^m$ are respectively position, velocity and acceleration of agent $i \in \mathcal{B}$ in the m-dimensional space at step k, and $dT \in \mathbb{R}^+$ is the time step. We consider physical constraints on velocities and accelerations, described by the sets V and A, respectively, which are defined by $V = \{v \mid |v| \leq \bar{v}\}$ and $A = \{a \mid |a| \leq \bar{a}\}$, where $\bar{v}$ and $\bar{a}$ limit the allowed magnitude of the velocity and acceleration vectors, respectively.

The *configuration* of all agents is described by the vector $\mathbf{x}(k) = [x_1^T(k)\ \ldots\ x_n^T(k)]^T \in \mathbb{R}^{m \cdot n}$. Let $\mathbf{v}(k) = [v_1^T(k)\ \ldots\ v_n^T(k)]^T \in \mathbb{R}^{m \cdot n}$, and $\mathbf{a}(k) = [a_1^T(k)\ \ldots\ a_n^T(k)]^T \in \mathbb{R}^{m \cdot n}$. Then the equation of motion for all agents can be expressed as

$$\mathbf{x}(k+1) = \mathbf{x}(k) + dT \cdot \mathbf{v}(k), \tag{3}$$

$$\mathbf{v}(k+1) = \mathbf{v}(k) + dT \cdot \mathbf{a}(k), \tag{4}$$

The local neighborhood of agent i is defined by the set of other agents, called *neighbors*, within a given distance from i, mimicking the agent's visibility sphere. For an *interaction radius* $r > 0$ and configuration $\mathbf{x}$, the set of *spatial neighbors* of agent i, $N_i(\mathbf{x}) \subseteq \mathcal{B}$, is given by:

$$N_i(\mathbf{x}) = \left\{ j \in \mathcal{B} \mid j \neq i \wedge \|x_i - x_j\| < r \right\}, \tag{5}$$

where $\|\cdot\|$ denotes the Euclidean norm.

For configuration $\mathbf{x} \in \mathbb{R}^{m \cdot n}$, we define the associated *proximity net* $G(\mathbf{x}) = (\mathcal{B}, \mathcal{E}(\mathbf{x}))$ as the graph that connects agents within their interaction radius:

$$\mathcal{E}(\mathbf{x}) = \left\{ (i,j) \in \mathcal{B} \times \mathcal{B} \mid \|x_i - x_j\| < r, i \neq j \right\}, \tag{6}$$

Definition 2.1 (α-lattice [4]). A configuration $\mathbf{x} \in \mathbb{R}^{m \cdot n}$ is called *α-lattice* if for all $i \in \mathcal{B}$ and all $j \in N_i(\mathbf{x})$, $\|x_i - x_j\| = d$, where $d \in \mathbb{R}^+$ is the scale of the α-lattice. For tolerance $\delta \in \mathbb{R}^+$, a configuration $\mathbf{x} \in \mathbb{R}^{m \cdot n}$ is called a *quasi α-lattice* if for all $i \in \mathcal{B}$ and all $j \in N_i(\mathbf{x})$, $|\|x_i - x_j\| - d| \leq \delta$.

2.1 Reynolds' rule-based model

In Reynolds' rule-based distributed model [7, 8], each agent $i \in \mathcal{B}$ updates its acceleration $a_i(k)$ at step k by considering the following three components :

(1) *Alignment*: agents match their velocities with the average velocity of nearby agents.
(2) *Cohesion*: agents move towards the centroid of the agents in the local neighborhood.
(3) *Separation*: agents move away from nearby neighbors.

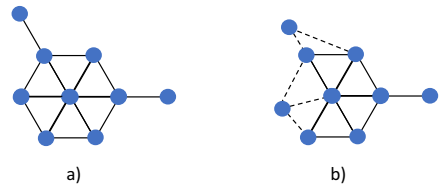


Figure 1: Examples of α-lattice a) and quasi α-lattice b). Solid lines connect agents in the same neighborhood that have distance d. Dashed lines connect those with have distance $d \pm \epsilon$ for $\epsilon \leq \delta$ (the tolerance).

2.2 Olfati-Saber's potential-based model

In potential-based flocking models, the interaction between a pair of agents is modeled by a potential field. It is assumed that an agent is a point source, and it has a potential field around it, which exerts a force, equal to its gradient, on other agents in its range of influence.In the work of Olfati-Saber [4], the potential function ψ_α for a pair of agents has its minimum at the desired inter-agent distance d of the desired α-lattice. Outside the interaction radius r, the potential function is constant, so the potential field exerts no force.

2.3 MPC-based models

Model predictive control (MPC) [1] is a well-established control technique that works as follows: at each time step k, it computes the optimal control sequence (agents' accelerations in our case) that minimizes a given cost function with respect to a predictive model of the controlled system and a finite prediction horizon of length T, i.e., from step $k+1$ to $k+T$. Then, the first control input of the optimal sequence is applied (the remainder of the sequence is unused), and the algorithm proceeds with a new iteration.

Two main kinds of MPC-based flocking models exist, *centralized* and *distributed*. Please refer to the full version of this paper on arxiv.org for further details.

3 DECLARATIVE FLOCKING

This section introduces centralized and distributed versions of our Declarative Flocking (DF) model, and presents a flocking algorithm based on MPC. Our formulation is declarative in that it consists of just two simple terms: (1) a cohesion term based on the average squared distance between pairs of agents, to keep the flock together, and (2) a separation term based on the inverse squared distances between pairs of agents, to avoid crowding. These two terms represent opposing forces on agents, causing agents to move towards positions in which these forces are balanced.

3.1 Centralized DF model

The cost function J for our centralized DF model contains the two terms described above, with the cohesion term considering all pairs of agents, and the separation term considering only pairs of agents that are neighbors. The weight ω of the separation term provides control over the density of the flock.

$$J^{\mathrm{C}}(\mathbf{x}) = \frac{2}{|\mathcal{B}| \cdot (|\mathcal{B}| - 1)} \cdot \sum_{i \in \mathcal{B}} \sum_{j \in \mathcal{B}, i<j} \|x_{ij}\|^2 + \omega \cdot \sum_{(i,j) \in \mathcal{E}(\mathbf{x})} \frac{1}{\|x_{ij}\|^2}$$

The control law is Eq. (??) with $J(k)$ equal to $\sum_{t=1}^{T} J^{\mathrm{C}}(\mathbf{x}(k+t \mid k))$.

3.2 Distributed DF model

The cost function J for our distributed DF model is similar to the centralized one, except that both terms are limited to consider pairs of agents that are neighbors.

$$J_i^{\mathrm{D}}(\mathbf{x}) = \frac{1}{|N_i(k)|} \cdot \sum_{j \in N_i(k)} \|x_{ij}\|^2 + \omega \cdot \sum_{j \in N_i(k)} \frac{1}{\|x_{ij}\|^2} \tag{7}$$

The control law for agent i is Eq. (??) with $J_i(k)$ equal to $\sum_{t=1}^{T} J_i^{\mathrm{D}}(\mathbf{x}(k+t \mid k))$.

4 MEASURES OF FLOCKING PERFORMANCE

We introduce four key measures of flocking performance. A single measure is insufficient, because flocking is indeed characterized by multiple desirable properties, such as aligned velocities and cohesion. Olfati-Saber introduces four main properties for flocking [4], informally described as:

(1) the group of agents stays *connected* in a unique flock, i.e., no sub-flocks and fragmentation should emerge;
(2) the group remains *cohesive*, in a close-knit formation;
(3) the group moves in a coherent way as if it was a unique body, i.e., agents' velocities are aligned; and
(4) the group maintains a regular geometry (in the α-lattice sense).

We introduce the following four measures to capture these four requirements. An important concept in these definitions is a *sub-flock*, which is a set of interacting agents that is too far apart from other agents to interact with them. Formally, a sub-flock in a configuration $\mathbf{x}$ corresponds to a connected component of the proximity net $G(\mathbf{x})$. Let $CC(\mathbf{x}) \subseteq 2^{\mathcal{B}}$ be the set of connected components of the proximity net $G(\mathbf{x})$.

(1) The *number of connected components* of the proximity net quantifies connectedness—or, equivalently, fragmentation—of the flock. There is no fragmentation when $|CC(\mathbf{x})| = 1$. Fragmentation exists when $|CC(\mathbf{x})| > 1$. Fragmentation may be temporary or, if sub-flocks move in different directions, permanent.

(2) The *maximum component diameter*, denoted $D(\mathbf{x})$, quantifies cohesion. It is defined by

$$D(\mathbf{x}) = \max_{\mathcal{B}' \in CC(\mathbf{x})} D(\mathbf{x}, \mathcal{B}') \tag{8}$$

where $D(\mathbf{x}, \mathcal{B}')$ is the diameter of connected component $\mathcal{B}'$:

$$D(\mathbf{x}, \mathcal{B}') = \max_{\substack{(i,j) \in \mathcal{B}' \times \mathcal{B}' \\ i \neq j}} \|\mathbf{x}_{ij}\|. \tag{9}$$

(3) The *velocity convergence* measure, adopted from [12], quantifies the average discrepancy between each agent's velocity and the average velocity of the flock. In particular, we extend the measure of [12] to average velocity convergence values across sub-flocks:

$$VC(\mathbf{x}, \mathbf{v}) = \frac{\sum_{\mathcal{B}' \in CC(\mathbf{x})} \left\| \sum_{i \in \mathcal{B}'} v_i - \left(\frac{\sum_{j \in \mathcal{B}'} v_j}{|\mathcal{B}'|} \right) \right\|^2 \Big/ |\mathcal{B}'|}{|CC(\mathbf{x})|} \tag{10}$$

(4) To measure the regularity of the geometric structure of a flock, as reflected in the inter-agent spacing, we introduce a parameter-free and model-independent *irregularity* measure $I(\mathbf{x})$. For a connected component (sub-flock) $\mathcal{B}'$, it is defined as the sample standard deviation of the distances between each agent in $\mathcal{B}'$ and its closest neighbor. Thus, the measure penalizes configurations where there is dispersion in inter-agent distances, while not imposing any fixed distance between them (unlike α-lattices).

Let $CC'(\mathbf{x}) = CC(\mathbf{x}) \setminus \bigcup_{i \in \mathcal{B}} \{\{i\}\}$ be the set of connected components where isolated agents are excluded. For $|CC'(\mathbf{x})| = 0$ (or equivalently, $|CC(\mathbf{x})| = |\mathcal{B}|$), i.e., all agents are isolated, we set the irregularity $I(\mathbf{x}) = 0$, which is the optimal value. This reflects the fact that a single point is a regular structure on its own. Moreover, such a configuration is already highly penalized by $|CC(\mathbf{x})|$ and $VC(\mathbf{v})$. For $|CC'(\mathbf{x})| > 0$, the measure is defined by:

$$I(\mathbf{x}) = \frac{\sum_{\mathcal{B}' \in CC'} \sigma\left(\biguplus_{i \in \mathcal{B}'} \min_{j \neq i} \|x_{ij}\| \right)}{|CC'|}. \tag{11}$$

where $\sigma(S)$ is the standard deviation of the multiset of samples S and $\biguplus$ is the sum operator (or disjoint union) for multisets.

An α-lattice (see Def. 2.1) has the optimal value of $I(\mathbf{x})$, i.e., $I(\mathbf{x}) = 0$, since all neighboring agents are located at the same distance d from each other, leading to zero standard deviation for the term $\sigma(\{d, d, \ldots, d\})$. This shows that $I(\mathbf{x})$ captures the regularity underlying the concept of α-lattice.

We introduce this measure because previous measures of regularity or irregularity, such as those in [4, 11, 12], measure deviations from an α-lattice with a specified inter-agent distance d and are therefore inapplicable to flocking models, such as Reynolds' model and our DF models, that are not based on α-lattices and do not have a specified target inter-agent distance. Also, our irregularity measure is more flexible than those based on α-lattices, because it gives an optimal score to some configurations that are geometrically regular but not α-lattices.

5 PERFORMANCE EVALUATION

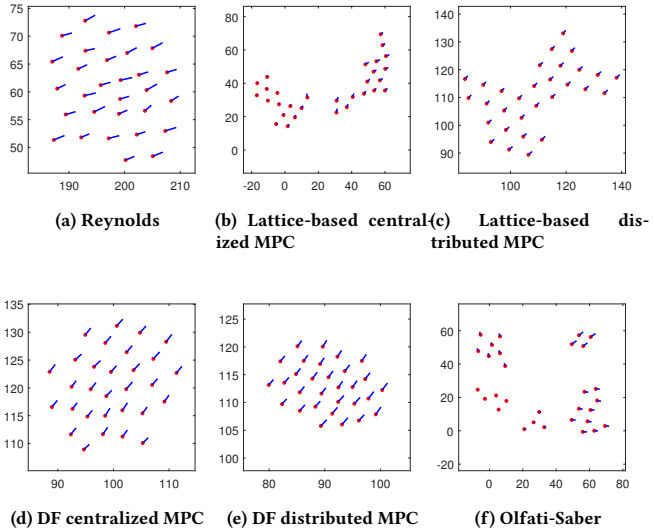
We compare the performance of the models of Section 2 with the newly introduced DF flocking models in the 2-dimensional setting. In the first set of experiments (Section 5.1), we evaluate the performance measures illustrated in Section 4. In the second set of experiments (Section 5.2), we analyze the resilience of the algorithms to sensor noise.

Unless otherwise specified, the population size is $n = 30$, the simulation length is 100, $dT = 0.3$, $\bar{v} = 8$, $\bar{a} = 1$, $r = 8.4$, $d = 7$, $T = 3$, and $\lambda = 1$. For further details about experimental settings please refer to the full version on arxiv.org.

5.1 Performance Comparison of Flocking Algorithms

Fig. 2 shows examples of final formations for all flocking models.

In Fig. 3, we compare the performance measures averaged over 100 runs for each flocking model. Regarding the number of connected components (sub-flocks), our centralized DF-MPC registers the best behavior, rapidly stabilizing to an average of 1 component (see plot a). Our distributed DF-MPC and Reynolds' model have



(a) Reynolds (b) Lattice-based centralized MPC (c) Lattice-based distributed MPC

(d) DF centralized MPC (e) DF distributed MPC (f) Olfati-Saber

Figure 2: Examples of final formations for different flocking models. The red dots are the agent positions. The blue lines denote the agent velocity vectors

comparable performance, reaching an average number of sub-flocks below 1.4. The lattice-based MPCs and Olfati-Saber instead lead to constant fragmentation, with more than 2 sub-flocks for the distributed lattice-based MPC, 6 for the centralized lattice-based MPC, and more than 8 for Olfati-Saber's model.

This ranking is confirmed by the diameter measure (plot b), where our centralized and distributed DF-MPC and Reynolds' model show the best cohesion, outperforming the lattice-based approaches. Recall that this measure indicates the maximum diameter over all sub-flocks, not the diameter of the entire population. As a consequence, fragmentation tends to improve diameter values since it produces sub-flocks with fewer individuals. This explains why our distributed DF-MPC performs better on this measure than the centralized version, and similarly why Olfati-Saber's model has smaller diameter measure than centralized lattice-based MPC, which in turn has smaller diameter measure than the distributed variant.

As expected, Olfati-Saber's model and the lattice-based MPCs have very good performance for irregularity (plot c), since they are designed to achieve the regular geometric formation of α-lattice. Surprisingly, our distributed DF-MPC performs almost as well as them on this measure. Centralized DF-MPC and Reynolds' model have the least regular formations.

For velocity convergence (plot d), we find that all models perform comparably well and are able to achieve flocks with consistent velocities fairly quickly after an initial spike.

5.2 Robustness to Sensing Noise

To evaluate the resiliency of the models to sensor noise, we performed 20 runs for each model at 10 noise levels. The noise levels are numbered from 1 to 10, and noise level i has $\sigma_x = 0.2i$ and $\sigma_v = 0.1i$. For each performance metric, we averaged its final values over 20 runs for each noise level. The results are plotted in Fig. 4.

Of the six models, Olfati-Saber's model is the most vulnerable to sensing noise: the number of sub-flocks $|CC|$ in Olfati-Saber's model quickly increases to nearly 30, rendering other metrics irrelevant. The lattice-based MPC models also exhibit high fragmentation, leading to nominally good but largely irrelevant values for the other performance metrics. Our distributed DF-MPC and Reynolds' model have the best resiliency to sensing noise, with both models exhibiting similar profiles in all metrics. While the irregularity and velocity convergence measures increase with noise level, as expected, both models remarkably maintain almost a single connected component with a nearly constant component diameter for all 10 noise levels, with DF-MPC achieving a smaller diameter than Reynolds' model.

6 RELATED WORK

Reynolds [7] introduced the first rule-based approach for simulation of flocking behavior. With three simple rules, his model is able to capture complex flocking behaviors of animals. Similar rule-base flocking models are also proposed by Pearce *et al.* [5] and Cucker and Dong [2].

Artificial potential fields have also been used extensively in flocking models. For example, Tanner *et al.* [9]. Ogren et.al. [6] use the motion of the leader to guide the motion of the flock; the leader's motion is independent.

La and Sheng [3] propose an extension of Olfati-Saber's model designed for noisy environments. In addition to the terms found in Olfati-Saber's model, their control law contains feedback terms for position and velocity, to make agents tend to stay close to the centroid of their neighborhood and minimizing the velocity mismatch with their neighbors. For further details regarding the related works please refer the full version of this paper on arxiv.org.

7 CONCLUSIONS

This paper presents an abstract declarative form of control for flocking behavior and the results of a thorough comparison of centralized and distributed versions of our MPC-based declarative flocking with four other flocking models. Our simulation results demonstrate that DF-MPC yields the best cohesion and least fragmentation, and produces natural flock shapes like those produced by Reynolds' rule-based model. Our resiliency analysis shows that the distributed version of our DF-MPC is highly robust to sensor noise.

As future work, we plan to study resilience of the flocking models with respect to additional noisy scenarios such as actuation noise (i.e., noise affecting acceleration) and faulty agents with deviant behavior. We also plan to investigate smoothing techniques to increase resilience to sensor noise.

REFERENCES

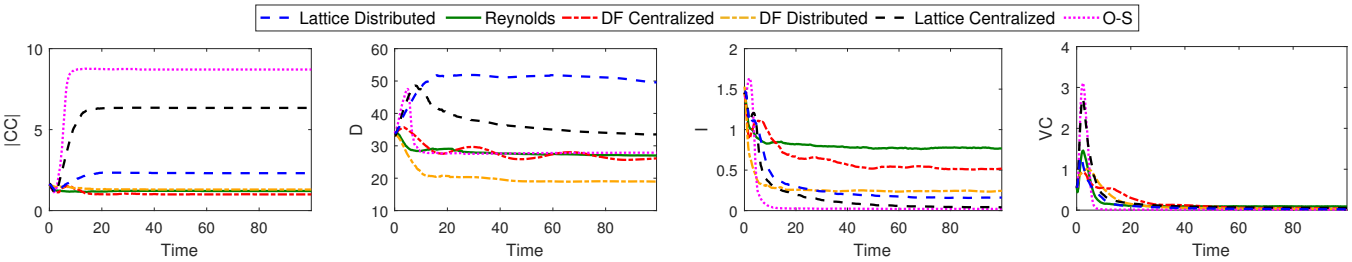
[1] E.F Camacho and C. Bordons. 2007. *Model predictive control.* Springer.

[2] Felipe Cucker and Jiu-Gang Dong. 2011. A general collision-avoiding flocking framework. *IEEE Trans. Automat. Control* 56, 5 (2011), 1124–1129.

[3] H. M. La and W. Sheng. 2010. Flocking control of multiple agents in noisy environments. In *2010 IEEE International Conference on Robotics and Automation.* 4964–4969. https://doi.org/10.1109/ROBOT.2010.5509668

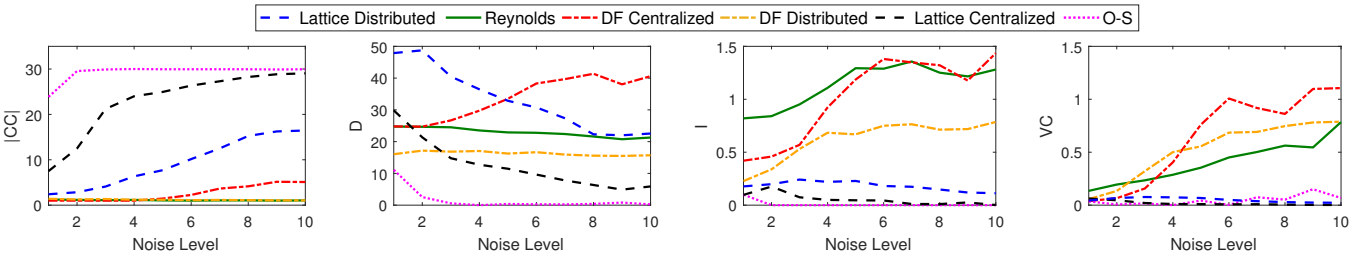
[4] Reza Olfati-Saber. 2006. Flocking for multi-agent dynamic systems: Algorithms and theory. *IEEE Transactions on automatic control* 51, 3 (2006), 401–420.

[5] Daniel J. G. Pearce, Adam M. Miller, George Rowlands, and Matthew S. Turner. 2014. Role of projection in the control of bird flocks. *Proceedings of the National*



(a) Number of connected components $|CC|$ **(b) Max component diameter D** **(c) Irregularity I** **(d) Velocity convergence VC**

Figure 3: Comparison of performance measures obtained with 100 runs for each flocking algorithm.



(a) Number of connected components $|CC|$ **(b) Max component diameter D** **(c) Irregularity I** **(d) Velocity convergence VC**

Figure 4: Comparison of the final values of the performance measures obtained with 20 runs for each flocking algorithm and for each noise level.

Academy of Sciences 111, 29 (2014), 10422–10426. https://doi.org/10.1073/pnas.1402202111 arXiv:http://www.pnas.org/content/111/29/10422.full.pdf

[6] Naomi Ehrich Leonard Peter Ogren. 2004. Cooperative control of mobile sensor networks:Adaptive gradient climbing in a distributed environment. *IEEE transactions on Automatic Control* 49, 8 (2004).

[7] Craig W. Reynolds. 1987. Flocks, Herds and Schools: A Distributed Behavioral Model. *SIGGRAPH Comput. Graph.* 21, 4 (Aug. 1987), 25–34. https://doi.org/10.1145/37402.37406

[8] Craig W. Reynolds. 1999. Steering Behaviors For Autonomous Characters. In *Proceedings of Game Developers Conference 1999*. 763–782.

[9] H. G. Tanner, A. Jadbabaie, and G. J. Pappas. 2003. Stable flocking of mobile agents part I: dynamic topology. In *42nd IEEE International Conference on Decision and Control (IEEE Cat. No.03CH37475)*, Vol. 2. 2016–2021 Vol.2.

[10] Jingyuan Zhan and Xiang Li. 2011. Flocking of discrete-time multi-agent systems with predictive mechanisms. *IFAC Proceedings Volumes* 44, 1 (2011), 5669–5674.

[11] Jingyuan Zhan and Xiang Li. 2013. Flocking of multi-agent systems via model predictive control based on position-only measurements. *IEEE Transactions on Industrial Informatics* 9, 1 (2013), 377–385.

[12] Hai-Tao Zhang, Zhaomeng Cheng, Guanrong Chen, and Chunguang Li. 2015. Model predictive flocking control for second-order multi-agent systems with input constraints. *IEEE Transactions on Circuits and Systems I: Regular Papers* 62, 6 (2015), 1599–1606.

[13] Lifeng Zhou and Shaoyuan Li. 2017. Distributed model predictive control for multi-agent flocking via neighbor screening optimization. *International Journal of Robust and Nonlinear Control* 27, 9 (2017), 1690–1705.