Flocking for Multi-Agent Dynamic Systems: Algorithms and Theory

Reza Olfati-Saber

Presented by: Daniel Geschwender

Overview

- Presents a framework for distributed flocking algorithms
  - Peer to peer architecture
  - Can handle obstacles and navigational goals
  - Embodies Reynolds’ three rules of flocking
- Three new algorithms for flocking
- 2D and 3D simulations using new algorithms

Outline

Day 1:

- Introduction
- Preliminaries
- Flocking Algorithms for Free-Space
- Collective Dynamics
- Stability Analysis

Day 2:

- Reynolds' Rules
- Flocking with Obstacle Avoidance
- Simulation Results
- What Constitutes Flocking?
- Conclusions

Outline

Day 1:

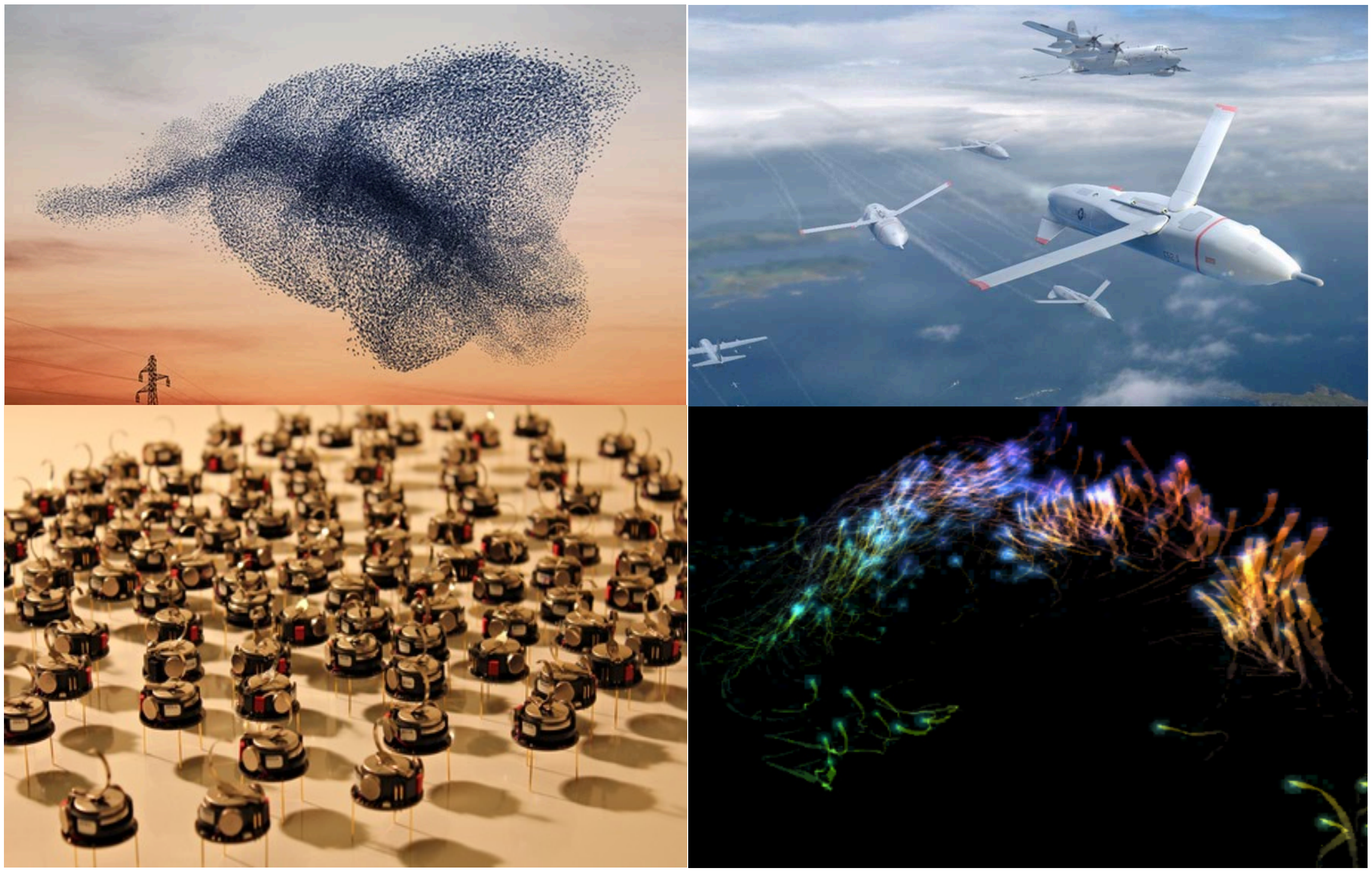
- Introduction
- Preliminaries
- Flocking Algorithms for Free-Space
- Collective Dynamics
- Stability Analysis

Day 2:

- Reynolds' Rules
- Flocking with Obstacle Avoidance
- Simulation Results
- What Constitutes Flocking?
- Conclusions

Introduction

- Applications of flocking:



Introduction

- Reynolds' three rules of flocking:
  - Flock Centering: attempt to stay close to nearby flockmates
  - Obstacle Avoidance: avoid collisions with nearby flockmates
  - Velocity Matching: attempt to match velocity with nearby flockmates

- Cohesion, separation, and alignment

Introduction

- Previous limitations include:
  1. The algorithms are centralized and require that each agent interacts with every other agent
  2. The algorithms do not possess (environmental) obstacle avoidance capabilities
  3. The algorithms lead to irregular fragmentation and/or collapse
  4. Unbounded forces are used for collision avoidance
  5. The algorithms do not possess distributed tracking (or migration) capabilities for groups

Introduction

- Fundamental questions:
  1. How do we design scalable flocking algorithms and guarantee their convergence?
  2. What does cohesion mean for groups and how is it achieved in a distributed way?
  3. What are the stability analysis problems related to flocking?
  4. What types of order exist in flocks?

Introduction

- Fundamental questions:

5. How do agents in flocks perform obstacle avoidance?

6. How do flocks perform split/rejoin maneuvers or pass through narrow spaces?

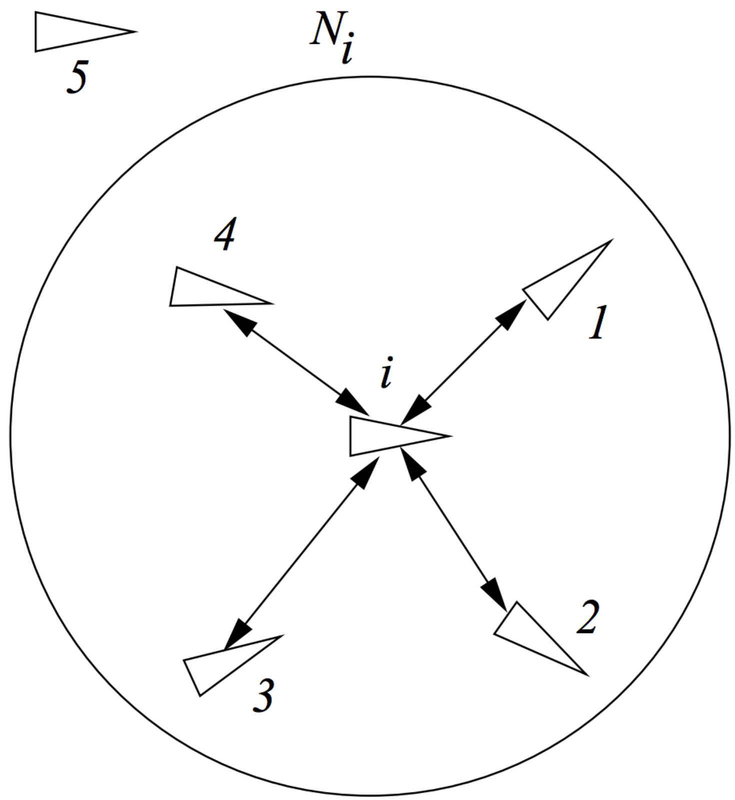
7. How do flocks migrate from point A to B? do they need any leaders?

8. What is a flock? and what constitutes flocking?

Preliminaries – graphs and nets

- Each agent has a position, $q_i \in \mathbb{R}^m$
- *Configuration* of all agents:
  $$q = \mathrm{col}(q_1, \ldots, q_n) \in Q = \mathbb{R}^{mn}$$
- *Framework*: configuration and corresponding graph (G, q)
- *Interaction range*, $r > 0$, defines the *spatial neighborhood* of each agent

Preliminaries – graphs and nets



$$N_i = \{j \in \mathcal{V} : \|q_j - q_i\| < r\}$$

Preliminaries – graphs and nets

- *Dynamic agents* have the following equations of motion:

$$\begin{cases} \dot{q}_i &= p_i, \\ \dot{p}_i &= u_i, \end{cases}$$

- In the dynamic case, the graph $G(q)$ may change over time and is referred to as *net*

Preliminaries – graphs and nets

- Interaction range limits agent communication to their local surroundings
- In this paper the nets are undirected
- Directed nets may be used if
  - Agents have different interaction ranges
  - Agents use a conic neighborhood

Preliminaries – α-lattices

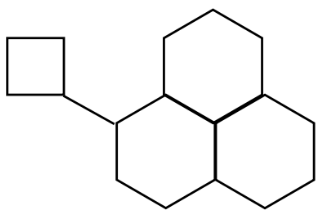
- An α*-lattice* is a configuration satisfying:

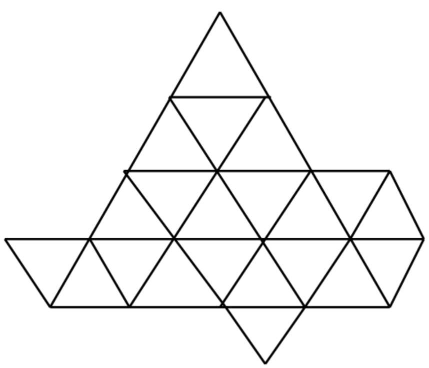
$$\|q_j - q_i\| = d, \quad \forall j \in N_i(q)$$

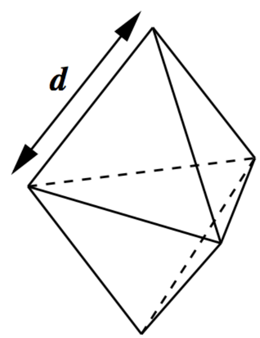
- *Scale* of an α-lattice: d
- *Ratio* of an α-lattice: $\kappa = r/d$

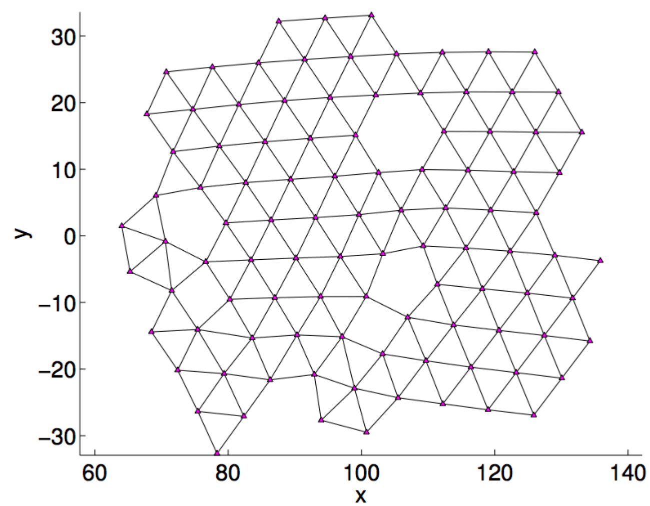
Preliminaries – α-lattices

- α-lattice examples:









Preliminaries – α-lattices

- A *quasi α-lattice* is a configuration satisfying:

$$-\delta \leq \|q_j - q_i\| - d \leq \delta, \quad \forall (i,j) \in \mathcal{E}(q)$$

- *Edge-length uncertainty*: $\delta \ll d$

Preliminaries – deviation energy

- *Deviation energy* measures how much a configuration differs from an α-lattice:

$$E(q) = \frac{1}{(|\mathcal{E}(q)| + 1)} \sum_{i=1}^{n} \sum_{j \in N_i} \psi(\|q_j - q_i\| - d)$$

- $\psi(z) = z^2$ is the pairwise potential function
- Deviation energy is a nonsmooth potential function

Preliminaries – σ-norm

- Euclidean norm is not differentiable at 0
- Instead, define the σ-norm (semi-norm):

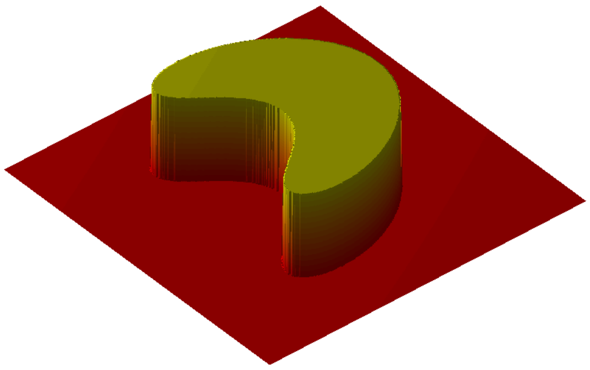
$$\|z\|_\sigma = \frac{1}{\epsilon}[\sqrt{1+\epsilon\|z\|^2}-1]$$

- The gradient is given by:

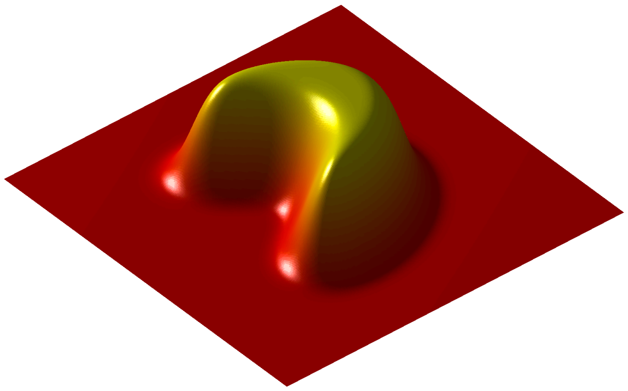
$$\sigma_\epsilon(z) = \nabla\|z\|_\sigma = \frac{z}{\sqrt{1+\epsilon\|z\|^2}} = \frac{z}{1+\epsilon\|z\|_\sigma}$$

Preliminaries – bump functions

- Indicator function vs. bump function



By Oleg Alexandrov - Own work, Public Domain, https://commons.wikimedia.org/w/index.php?curid=2531841



By Oleg Alexandrov - self-made with MATLAB, source code below, Public Domain, https://commons.wikimedia.org/w/index.php?curid=2819733

- Smooth adjacency matrices using:

$$\rho_h(z) = \begin{cases} 1, & z \in [0, h) \\ \frac{1}{2}[1 + \cos(\pi \frac{(z-h)}{(1-h)})], & z \in [h, 1] \\ 0 & \text{otherwise.} \end{cases}$$

Preliminaries – bump functions

- Define *spatial adjacency matrix* by:

$$a_{ij}(q) = \rho_h(\|q_j - q_i\|_\sigma / r_\alpha) \in [0, 1], \quad j \neq i$$

- Set $h = 1$, $\rho_h(z)$ becomes an indicator function, but is still differentiable

Preliminaries – collective potential

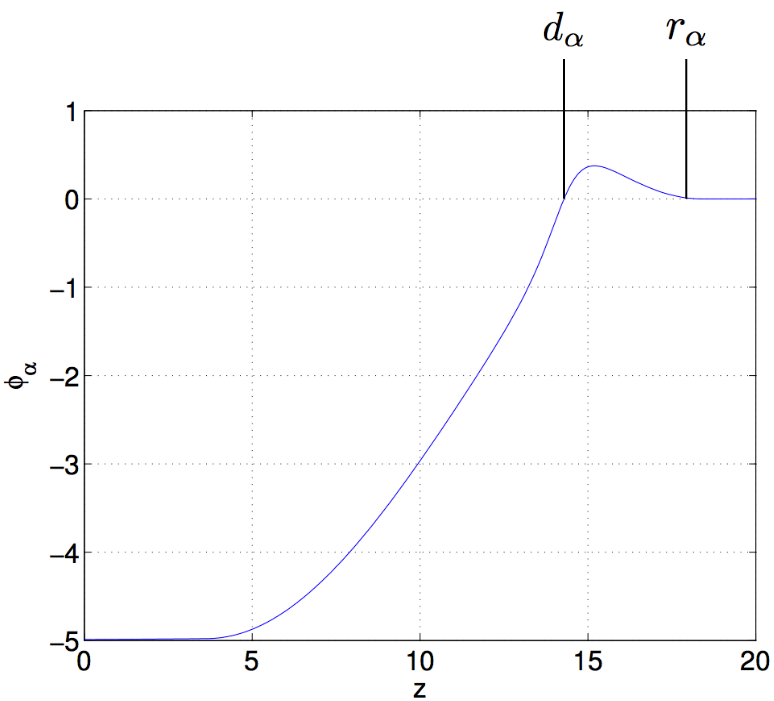
- Collective potential is a smooth version of deviation energy

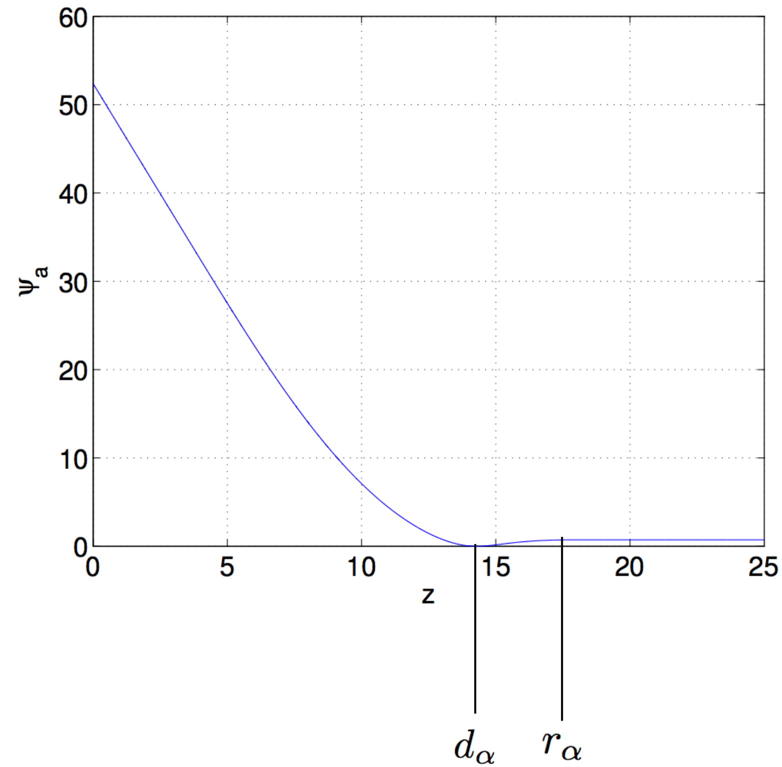
$$V(q) = \frac{1}{2} \sum_{i} \sum_{j \neq i} \psi_\alpha(\|q_j - q_i\|_\sigma)$$

$$\psi_\alpha(z) = \int_{d_\alpha}^{z} \phi_\alpha(s) ds.$$

$$\begin{aligned} \phi_\alpha(z) &= \rho_h(z/r_\alpha)\phi(z - d_\alpha) \\ \phi(z) &= \tfrac{1}{2}[(a+b)\sigma_1(z+c) + (a-b)] \end{aligned}$$

Preliminaries – collective potential





Preliminaries – Velocity Matching

- *Velocity mismatch function:*

$$K = \frac{1}{2}\sum_i \|v_i\|^2$$

- *Damping force:*

$$\dot{K} = -\frac{1}{2}\sum_{(i,j)\in\mathcal{E}} a_{ij}\|v_j - v_i\|^2 \leq 0$$

- Expressed with *m-dimensional graph Laplacians:*

$$z^T \hat{L} z = \frac{1}{2}\sum_{(i,j)\in\mathcal{E}} a_{ij}\|z_j - z_i\|^2, \quad z \in \mathbb{R}^{mn}$$

$$L = \Delta(A) - A \qquad \hat{L} = L \otimes \mathbf{1}_m$$

Free-space Flocking Algorithms

- Flocking without obstacles
- Two types of agents:
  - α-agents: flock members
  - γ-agents: group navigational objectives
- α-agent control input consists of 3 terms:

$$u_i = f_i^g + f_i^d + f_i^\gamma$$

Gradient based term

Velocity consensus term

Navigational feedback term

Free-space Flocking Algorithms

- Algorithm 1:

$$u_i^\alpha = \underbrace{\sum_{j \in N_i} \phi_\alpha(\|q_j - q_i\|_\sigma)\mathbf{n}_{ij}}_{\text{Gradient based term}} + \underbrace{\sum_{j \in N_i} a_{ij}(q)(p_j - p_i)}_{\text{Velocity consensus term}}$$

$$\mathbf{n}_{ij} = \sigma_\epsilon(q_j - q_i) = \frac{q_j - q_i}{\sqrt{1 + \epsilon\|q_j - q_i\|^2}}$$

Free-space Flocking Algorithms

- Algorithm 2:

$$u_i = \underbrace{\sum_{j\in N_i} \phi_\alpha(\|q_j - q_i\|_\sigma)\mathbf{n}_{ij}}_{\text{Gradient based term}} + \underbrace{\sum_{j\in N_i} a_{ij}(q)(p_j - p_i)}_{\text{Velocity consensus term}} + \underbrace{f_i^\gamma(q_i, p_i)}_{\text{Navigational feedback term}}$$

$$u_i^\gamma := f_i^\gamma(q_i, p_i, q_r, p_r) = -c_1(q_i - q_r) - c_2(p_i - p_r), \quad c_1, c_2 > 0.$$

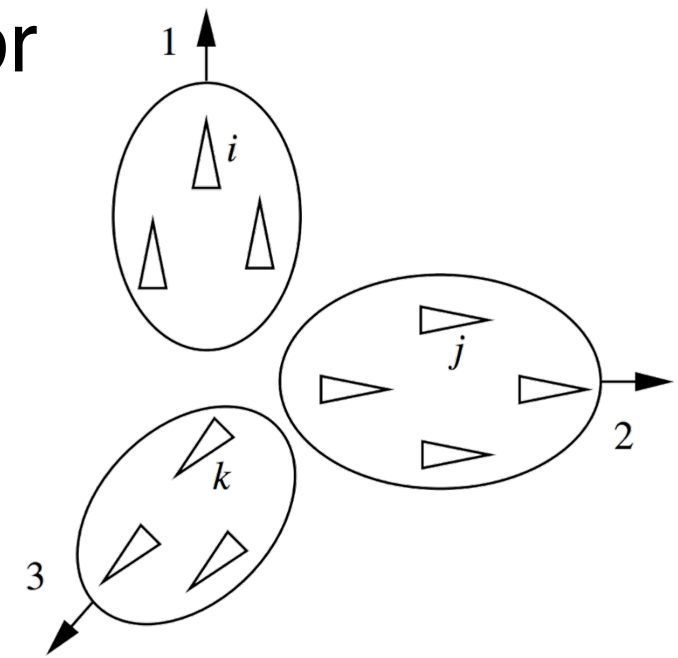
- State of γ-agents:

$$(q_r, p_r) \in \mathbb{R}^m \times \mathbb{R}^m$$

$$\begin{cases} \dot{q}_r &= p_r, \\ \dot{p}_r &= f_r(q_r, p_r), \end{cases}$$

Free-space Flocking Algorithms

- Algorithms 1 and 2 result in drastically different behavior
- Algorithm 1 embodies all 3 Reynolds' rules but rarely results in flocking
- Instead, it often creates fragmentation
- Algorithm 2 creates flocking



Collective Dynamics

- Collective dynamics of α-agents given by:
$$\begin{cases} \dot{q} &= & p \\ \dot{p} &= & -\nabla V(q) - \hat{L}(q)p + f_\gamma(q, p, q_r, p_r) \end{cases}$$

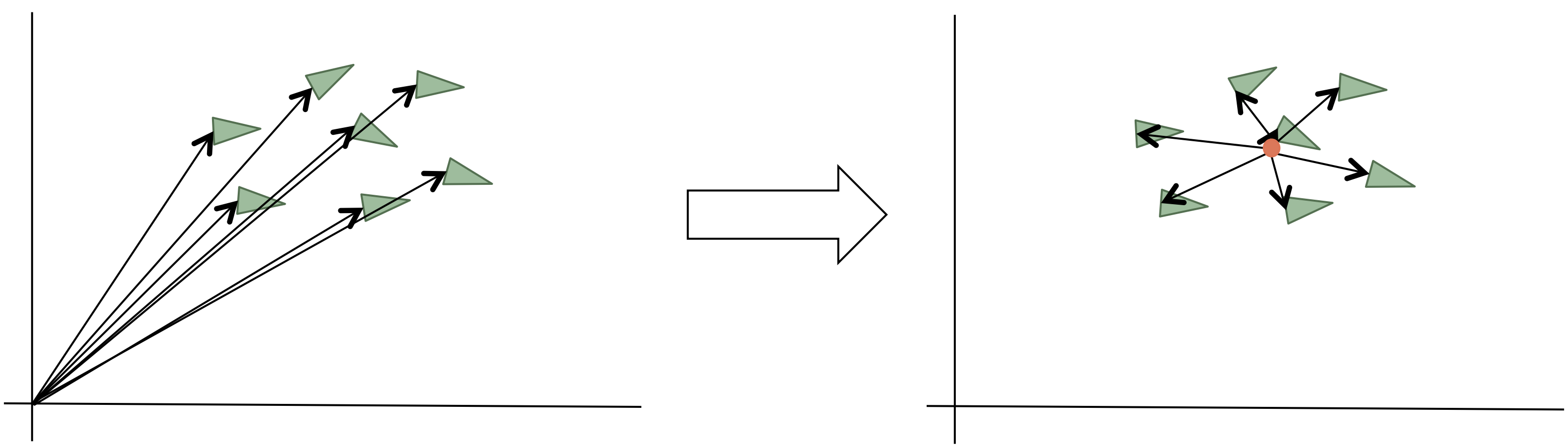
- Hamiltonian of the system (sum of all kinetic and potential energy):

$$H(q,p) = V(q) + \sum_{i=1}^{n} \|p_i\|^2.$$

Collective Dynamics

- Analyze the system from a moving frame centered on the center of mass, q_c

$$\begin{cases} x_i = q_i - q_c, \\ v_i = p_i - p_c. \end{cases}$$



Collective Dynamics

- Collective dynamics can be decomposed:
  - n systems in the moving frame

$$\textit{structural dynamics:} \begin{cases} \dot{x} &= v \\ \dot{v} &= -\nabla V(x) - \hat{L}(x)v + g(x,v) \end{cases}$$

  - 1 system in the reference frame

$$\textit{translational dynamics:} \begin{cases} \dot{q}_c &= p_c \\ \dot{p}_c &= h(q_c, p_c, q_r, p_r) \end{cases}$$

Stability Analysis

- Stable flocking motion analyzed in both:
  - Stability of certain equilibria of the structural dynamics
  - Stability of a desired equilibrium of the translational dynamics
- Animal behavior may not require translational stability but engineering applications do

Stability Analysis

- *Cohesive group*: All agents stay within a ball of radius R > 0 centered at q_c
- *Flock:* net G(q) is connected
- *Quasi-flock:* G(q) has a giant component
- *Dynamic flock:* agents are a flock over a given time interval

Stability Analysis

Theorem 1. *Consider a group of α-agents applying protocol* (24) *(Algorithm 1) with structural dynamics Σ_1 (defined in* (36)*). Let $\Omega_c = \{(x, v) : H(x, v) \le c\}$ be a level-set of the Hamiltonian $H(x, v)$ of Σ_1 such that for any solution starting in Ω_c, the group of agents is a cohesive dynamic flock for all $t \ge 0$. Then, the following statements hold:*

i) Almost every solution of the structural dynamics converges to an equilibrium $(x^*, 0)$ with a configuration x^* that is an α-lattice.

ii) The velocity of all agents asymptotically match in the reference frame.

iii) Given $c < c^* = \psi_\alpha(0)$, no inter-agent collisions occur for all $t \ge 0$.

- For algorithm 1
- Assumes cohesion
- proves: converges to α-lattice, velocities match, no collisions

Stability Analysis

Theorem 3. *Consider a group of α-agents applying protocol* (26) *(Algorithm 2) with $c_1, c_2 > 0$ and structural dynamics Σ_2 (defined in* (37)*). Assume that the initial velocity mismatch $K(v(0))$ and inertia $J(x(0))$ are finite. Then, the following statements hold:*

i) The group of agents remain cohesive for all $t \geq 0$.

ii) Almost every solution of Σ_2 asymptotically converges to an equilibrium point $(x_\lambda^*, 0)$ where x_λ^* is a local minima of $U_\lambda(x)$.

iii) The velocity of all agents asymptotically match in the reference frame.

iv) Assume the initial structural energy of the particle system is less than $(k+1)c^*$ with $c^* = \psi_\alpha(0)$ and $k \in \mathbb{Z}_+$. Then, at most k distinct pairs of α-agents could possibly collide ($k = 0$ guarantees a collision-free motion).

- For algorithm 2
- Does not assume cohesion
- proves: cohesion, converges to α-lattice, velocities match, no collisions

Stability Analysis

Theorem 5. *Let q be an α-lattice of scale $d > 0$ and ratio $\kappa > 1$ with n nodes at distinct positions. Then*

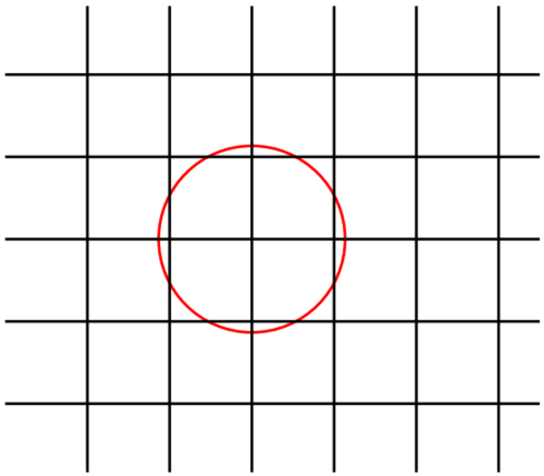
i) The structure $(G(q), q)$ induced by q is a planar graph in dimensions $m = 2, 3$.

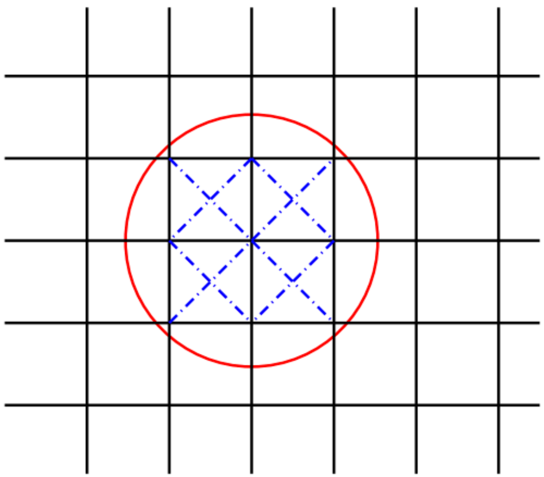
ii) The net $G(q)$ has at most $3n - 6$ links in dimension 2.

iii) The net $G(q)$ induced by an α-lattice with $n > m+1$ nodes cannot be a complete graph in dimensions $m = 1, 2, 3$.

- Edges of α-lattice do not cross
- # of edges is linear in the number of agents
- α-lattices cannot be complete for reasonably sized number of agents

Stability Analysis





- A given α-lattice may only exist under certain interaction ranges

Outline

Day 1:

- Introduction
- Preliminaries
- Flocking Algorithms for Free-Space
- Collective Dynamics
- Stability Analysis

Day 2:

- Reynolds' Rules
- Flocking with Obstacle Avoidance
- Simulation Results
- What Constitutes Flocking?
- Conclusions

Outline

Day 1:

- Introduction
- Preliminaries
- Flocking Algorithms for Free-Space
- Collective Dynamics
- Stability Analysis

Day 2:

- Reynolds' Rules
- Flocking with Obstacle Avoidance
- Simulation Results
- What Constitutes Flocking?
- Conclusions

Algorithm 1 vs. Reynolds Rules

- Algorithm 1 embodies all 3 rules:
  - Flock centering
  - Obstacle avoidance
  - Velocity matching
- Can be analyzed using *stress elements of a graph:*

$$s_{ij}(q) = \frac{\phi_\alpha(\|q_j - q_i\|_\sigma)}{1 + \epsilon\|q_j - q_i\|_\sigma}, \quad (i,j) \in \mathcal{E}(q)$$

Algorithm 1 vs. Reynolds Rules

- Algorithm 1 expressed in terms of stress and adjacency elements:

$$u_i^\alpha = \sum_{j \in N_i(q)} s_{ij}(q)(q_j - q_i) + \sum_{j \in N_i(q)} a_{ij}(q)(p_j - p_i) = u_i^g + u_i^{vm}$$

- First term encompasses flock centering and obstacle avoidance
- Second term encompasses velocity matching

Algorithm 1 vs. Reynolds Rules

- To further decompose the stress term, partition neighbors into positive, negative, and neutral friends:

$$F_i^+ = \{j \in N_i(q) : s_{ij}(q) > 0\},\ F_i^- = \{j \in N_i(q) : s_{ij}(q) < 0\},\ F_i^0 = \{j \in N_i(q) : s_{ij}(q) = 0\}.$$

- Move towards center of positive friends, away from center of negative friends:

$$\begin{aligned} u_i^g &= \sum_{j \in N_i(q)} s_{ij}(q) q_j - S_i(q) q_i \\ &= \sum_{j \in F_i^+} s_{ij}(q) q_j + \sum_{j \in F_i^-} s_{ij}(q) q_j \\ &= S_i^+(q)\langle q_i \rangle^+ + S_i^-(q)\langle q_i \rangle^- \end{aligned}$$

Flocking with Obstacle Avoidance

- Represent nearby (active) obstacles as agents: β-agents
- β-agents are induced when α-agents come into close proximity with obstacles
- Obstacles are restricted to connected convex regions
- Focus on spheres or infinite walls

Flocking with Obstacle Avoidance

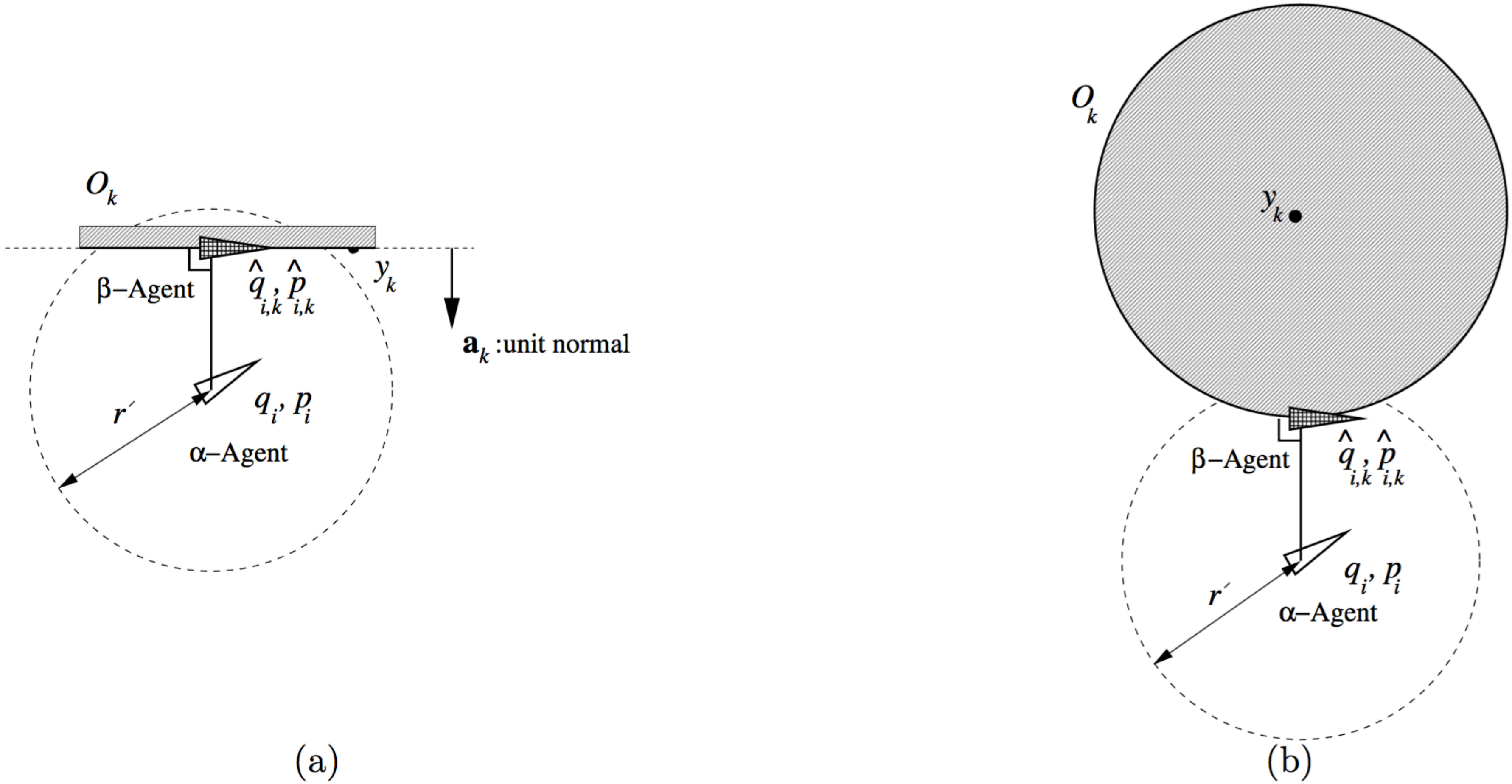


Figure 6: Agent-based representation of obstacles: (a) a wall and (b) a spherical obstacle.

Flocking with Obstacle Avoidance

- A β-agent is generated on the surface of obstacle, *k*, at the closest point to the α-agent, *i*:

$$\hat{q}_{i,k} = \text{argmin}_{x \in O_k} \|x - q_i\|$$

- β-agents are assigned a velocity based on their generating α-agent:

$$\hat{p}_{i,k} = \mu P p_i$$

$$P = I - \mathbf{a}_k \mathbf{a}_k^T$$

Flocking with Obstacle Avoidance

- α and β neighbors are distinguished:

$$
\begin{aligned}
N_i^\alpha &= \{j \in \mathcal{V}_\alpha : \|q_j - q_i\| < r\} \\
N_i^\beta &= \{k \in \mathcal{V}_\beta : \|\hat{q}_{i,k} - q_i\| < r'\}
\end{aligned}
$$

- Similarly, α and β edges are distinguished:

$$
\begin{aligned}
\mathcal{E}_\alpha(q) &= \{(i,j) : i \in \mathcal{V}_\alpha, j \in N_i^\alpha\}, \\
\mathcal{E}_\beta(q) &= \{(i,k) : i \in \mathcal{V}_\alpha, k \in N_i^\beta\},
\end{aligned}
$$

Flocking with Obstacle Avoidance

- A constrained α-lattice is an α-lattice with β-agents included.

$$\begin{cases} \|q_j - q_i\| & = & d, \ \forall j \in N_i^{\alpha} \\ \|\hat{q}_{i,k} - q_i\| & = & d', \ \forall k \in N_i^{\beta} \end{cases}$$

- Distances between two α-agents may differ from the distances between α and β-agents

Flocking with Obstacle Avoidance

- Multi-species collective potential function:

$$V(q) = c_1^{\alpha} V_{\alpha}(q) + c_1^{\beta} V_{\beta}(q) + c_1^{\gamma} V_{\gamma}(q)$$

$$\begin{aligned}
V_{\alpha}(q) &= \sum_{i \in \mathcal{V}_{\alpha}} \sum_{j \in \mathcal{V}_{\alpha} \setminus \{i\}} \psi_{\alpha}(\|q_j - q_i\|_{\sigma}), \\
V_{\beta}(q) &= \sum_{i \in \mathcal{V}_{\alpha}} \sum_{k \in N_i^{\beta}} \psi_{\beta}(\|\hat{q}_{i,k} - q_i\|_{\sigma}), \\
V_{\gamma}(q) &= \sum_{i \in \mathcal{V}_{\alpha}} (\sqrt{1 + \|q_i - q_r\|^2} - 1).
\end{aligned}$$

Flocking with Obstacle Avoidance

- Repulsive action function:

$$\phi_\beta(z) = \rho_h(z/d_\beta)(\sigma_1(z - d_\beta) - 1)$$

$$\sigma_1(z) = z/\sqrt{1+z^2}.$$

- Equals 0 for $z \geq d_\beta$, bounded
- Repulsive pairwise potential:

$$\psi_\beta(z) = \int_{d_\beta}^{z} \phi_\beta(s)ds \geq 0$$

Flocking with Obstacle Avoidance

- Algorithm 3:

$$u_i = u_i^\alpha + u_i^\beta + u_i^\gamma$$

$$\begin{aligned}
u_i^\alpha &= c_1^\alpha \sum_{j\in N_i^\alpha} \phi_\alpha(\|q_j - q_i\|_\sigma)\mathbf{n}_{i,j} + c_2^\alpha \sum_{j\in N_i^\alpha} a_{ij}(q)(p_j - p_i) \\
u_i^\beta &= c_1^\beta \sum_{k\in N_i^\beta} \phi_\beta(\|\hat{q}_{i,k} - q_i\|_\sigma)\hat{\mathbf{n}}_{i,k} + c_2^\beta \sum_{j\in N_i^\beta} b_{i,k}(q)(\hat{p}_{i,k} - p_i) \\
u_i^\gamma &= -c_1^\gamma \sigma_1(q_i - q_r) - c_2^\gamma (p_i - p_r)
\end{aligned}$$

Flocking with Obstacle Avoidance

Theorem 6. *Consider a particle system applying Algorithm 3 (or protocol (67)). Assume that the γ-agent is a static agent with a fixed state $(q_r, p_r) = (q_d, p_d)$. Define the energy function $H(q,p) = V(q) + T(q,p)$ with kinetic energy $T(q,p) = \frac{1}{2}\sum_{i=1}^{n} \|p_i\|^2$. Suppose there exists a finite time $t_0 \geq 0$ such that the average velocity of all agents satisfies the condition*

$$\frac{n}{2}\langle p_c(t), p_d\rangle \leq T(q(t), p(t)), \quad \forall t \geq t_0. \tag{76}$$

Then, the energy of the system is monotonically decreasing (i.e. $\dot{H}(q(t), p(t)) \leq 0$) along the trajectory of the collective dynamics of the multi-species system for all $t \geq t_0$.

- Take the derivative of the Hamiltonian and show that it is always negative:

$$\dot{H}(q,p) = -c_2^{\alpha}(p^T \hat{L}(q)p) + c_2^{\beta} \sum_{i \in \mathcal{V}_\alpha} \sum_{k \in N_i^{\beta}} b_{i,k}\langle p_i, \hat{p}_{i,k} - p_i\rangle - 2c_2^{\gamma}(T(q,p) - \frac{n}{2}(p_d^T \cdot p_c)) \leq 0, \quad \forall t \geq t_0$$

- Don’t analyze case with moving goal or permanent obstacles

Flocking with Obstacle Avoidance

- γ-agent may be considered a *virtual-leader*
- Is not a physical agent, and may be duplicated per α-agent for peer-to-peer

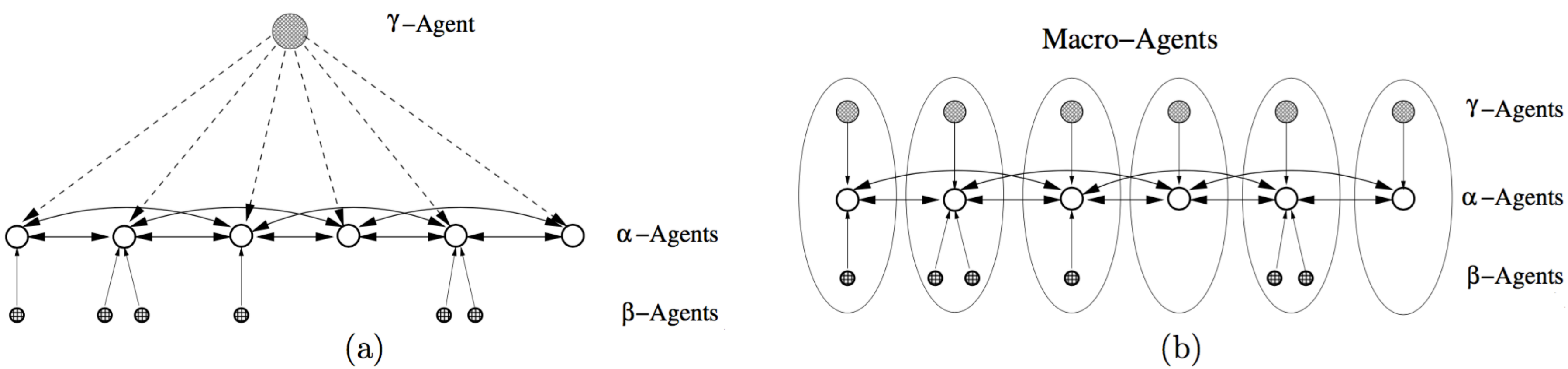


Figure 7: The in-agent and intra-agent information flow in constrained flocking: (a) a virtual-leader/follower hierarchical architecture and (b) a peer-to-peer architecture.

Flocking with Obstacle Avoidance

- Algorithm 3 attempts to take a direct route
- May be blocked by some obstacles

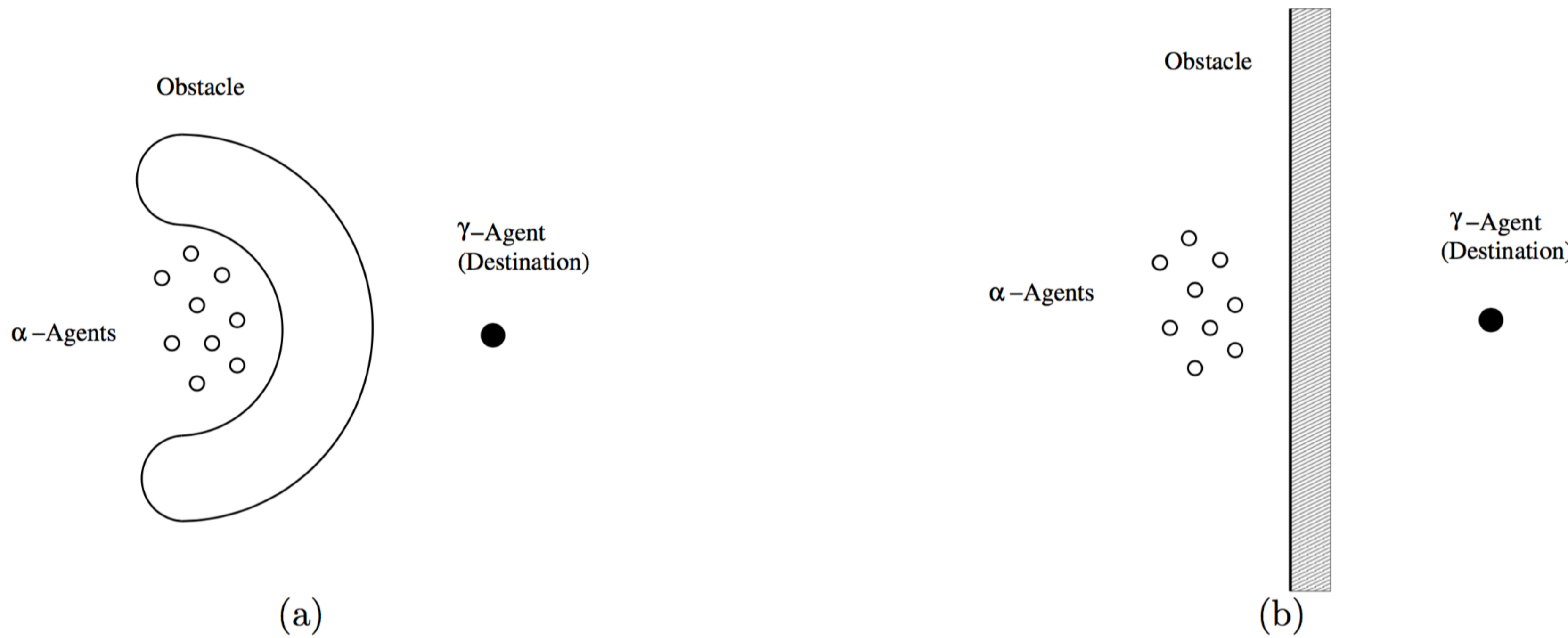


Figure 9: Trapped α-agents due to conflicting objectives: (a) a non-convex obstacle and (b) a convex obstacle.

Simulation Results

- Parameters:
  - d=7, r=1.2d, d’=0.6d, r’=1.2d’
  - ε=0.1 (for σ-norm)
  - h=0.2 (for $\phi_\alpha(z)$)
  - h=0.9 (for $\phi_\beta(z)$)
  - Step size between 0.01-0.03 seconds

Simulation Results

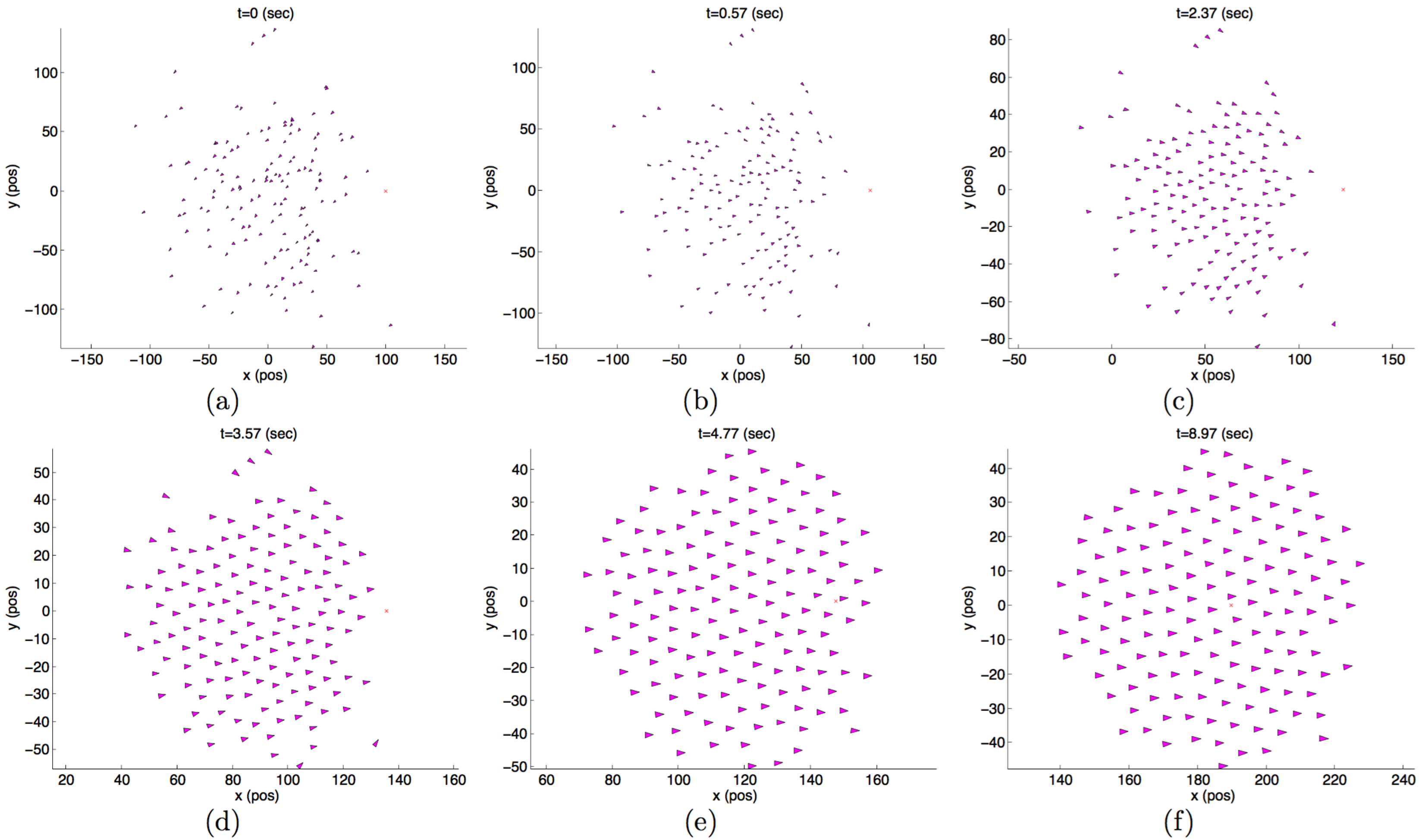


Figure 10: 2-D flocking for $n = 150$ agents.

Simulation Results

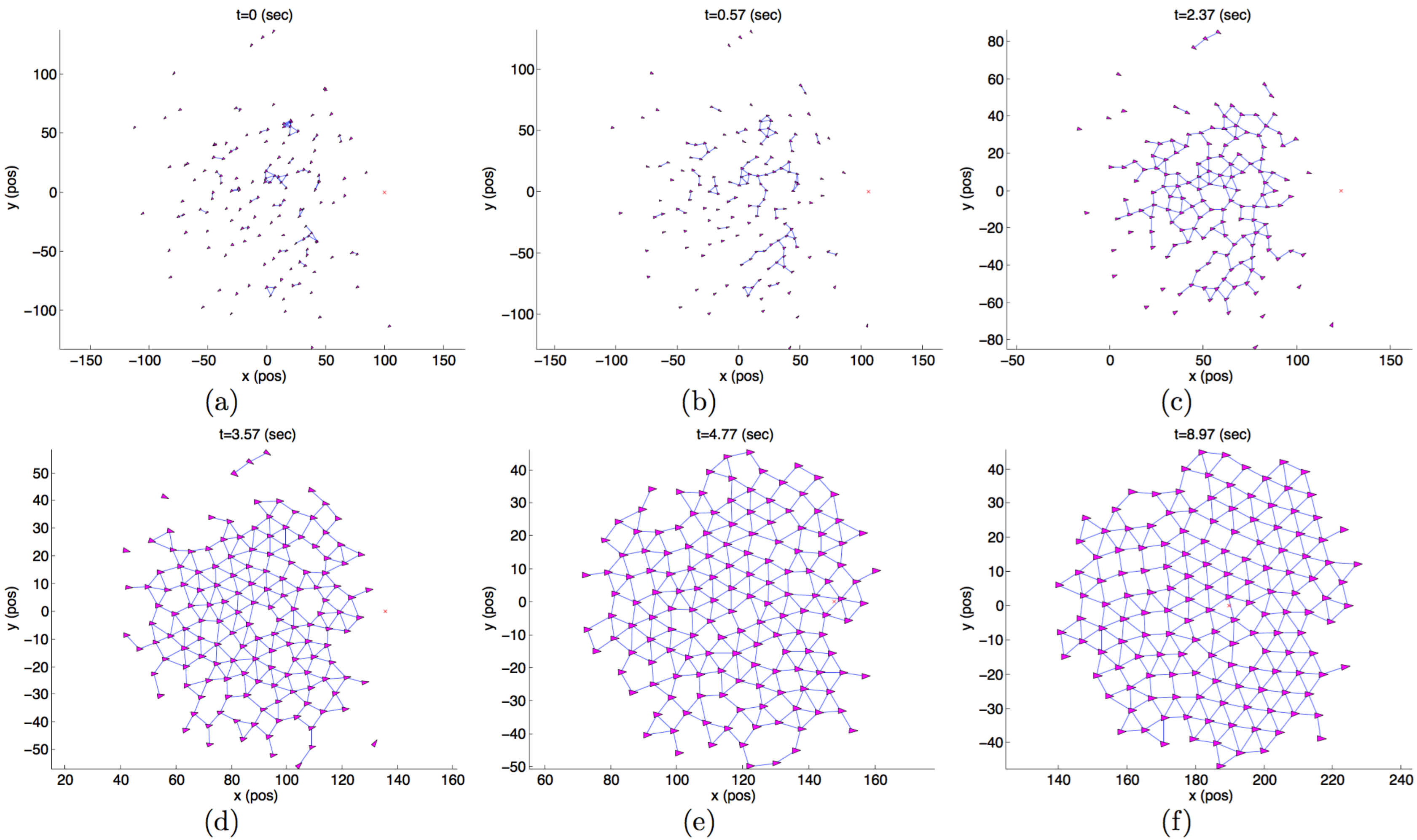


Figure 11: 2-D flocking for $n = 100$ with a dynamic topology.

Simulation Results

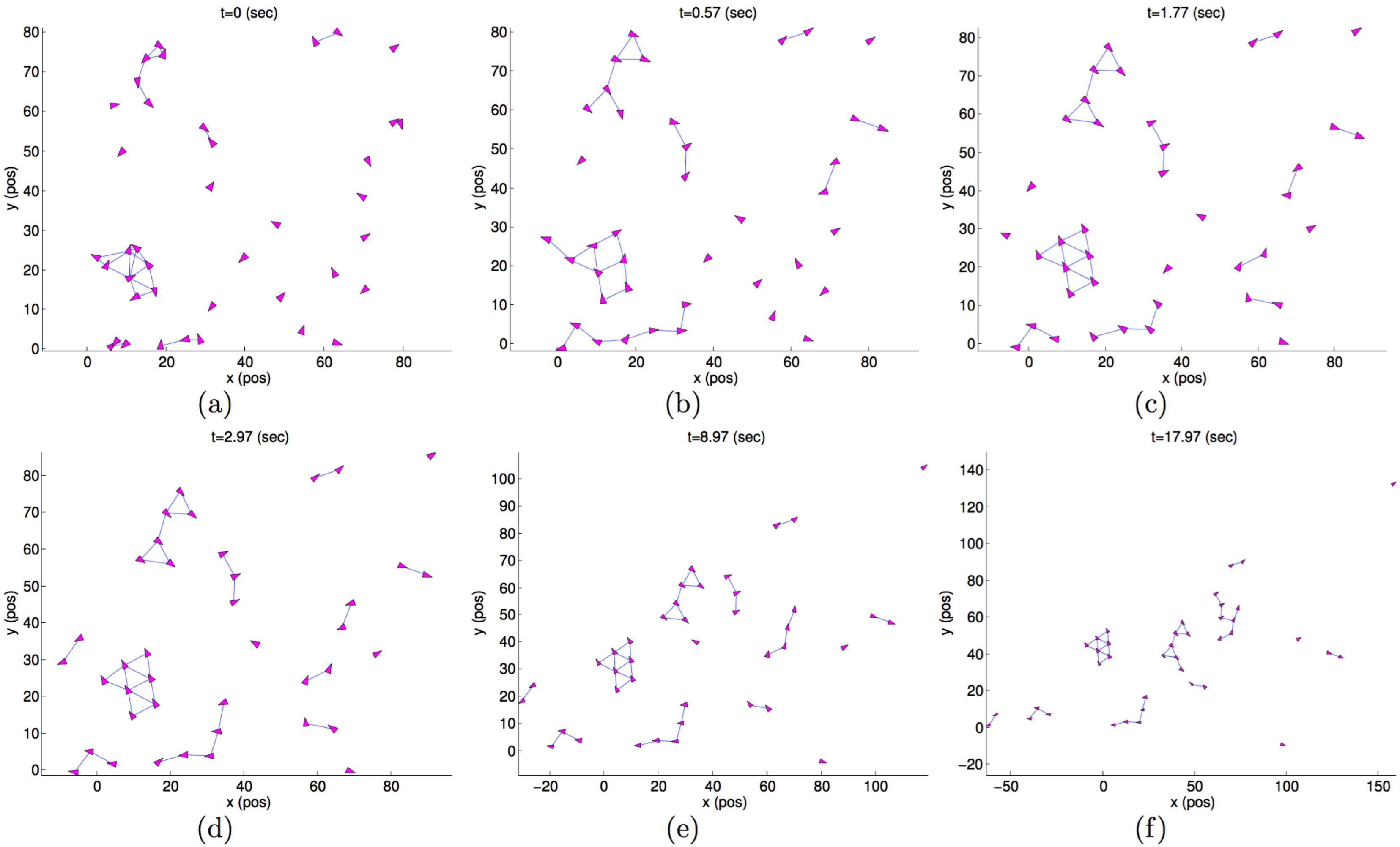


Figure 12: The fragmentation phenomenon for 40 agents applying Algorithm 1.

Simulation Results

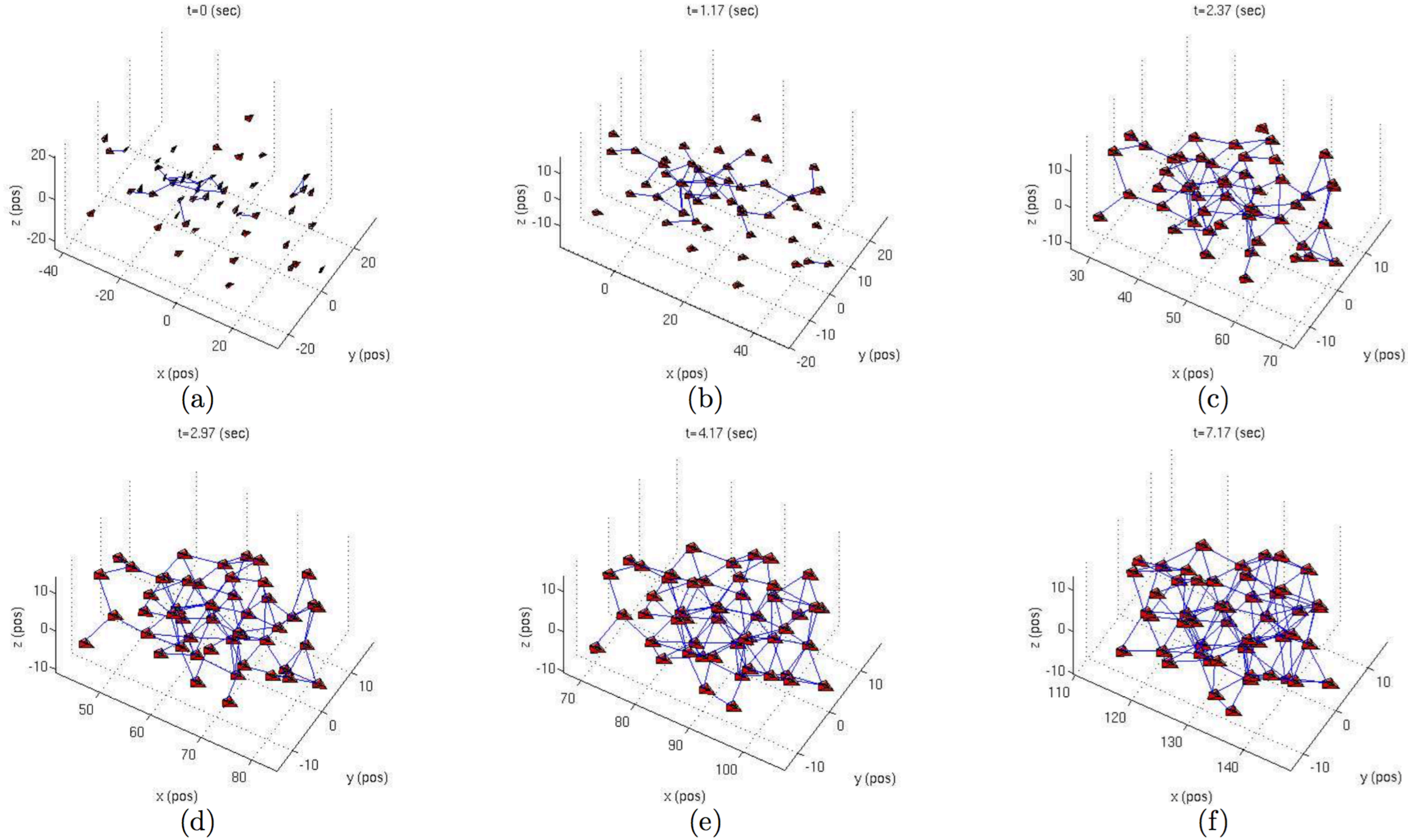


Figure 13: Snapshots of 3-D flocking/automated rendezvous using Algorithm 2 for $n = 50$ UAVs.

Simulation Results

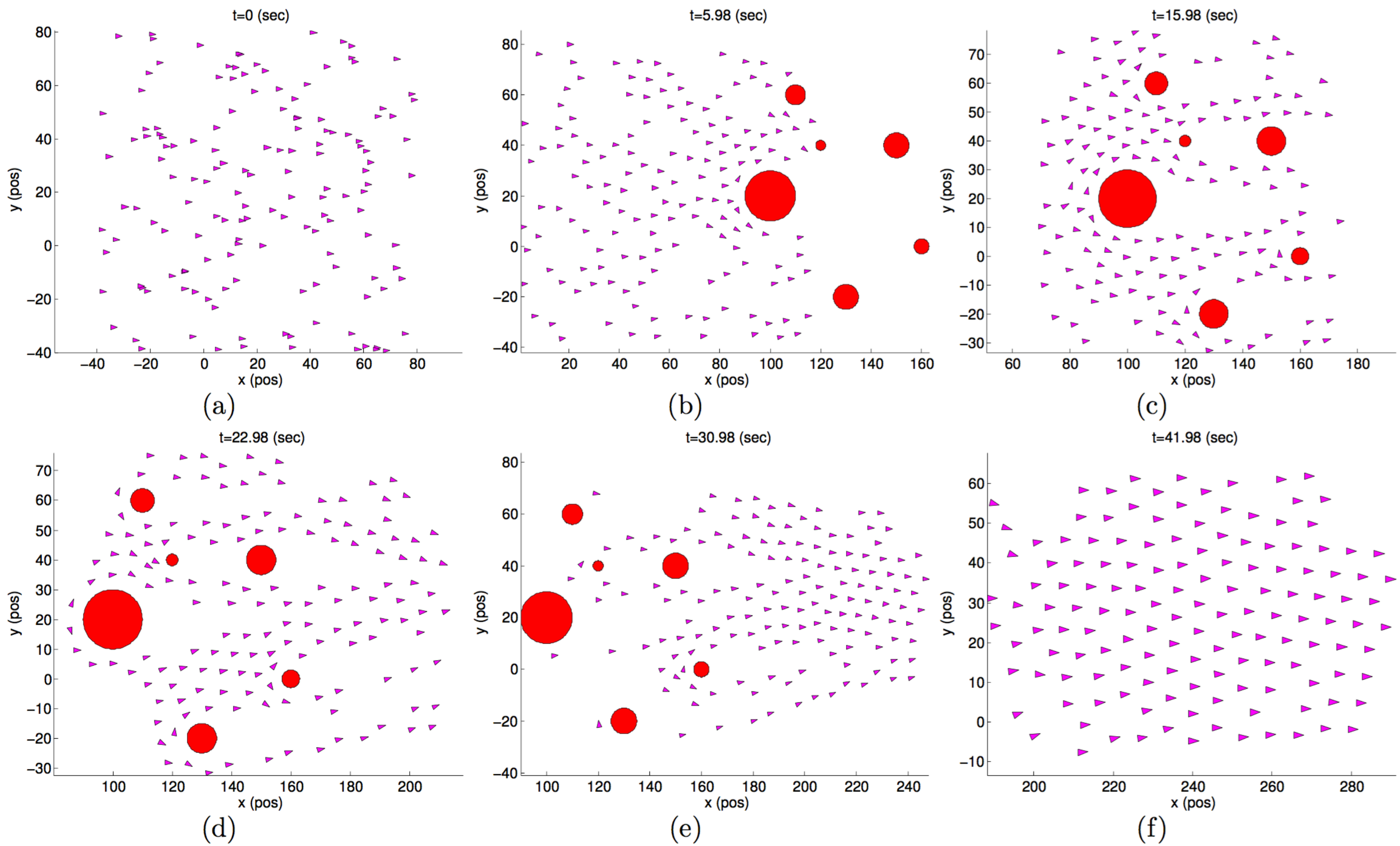


Figure 14: The split/rejoin maneuver for $n = 150$ agents.

Simulation Results

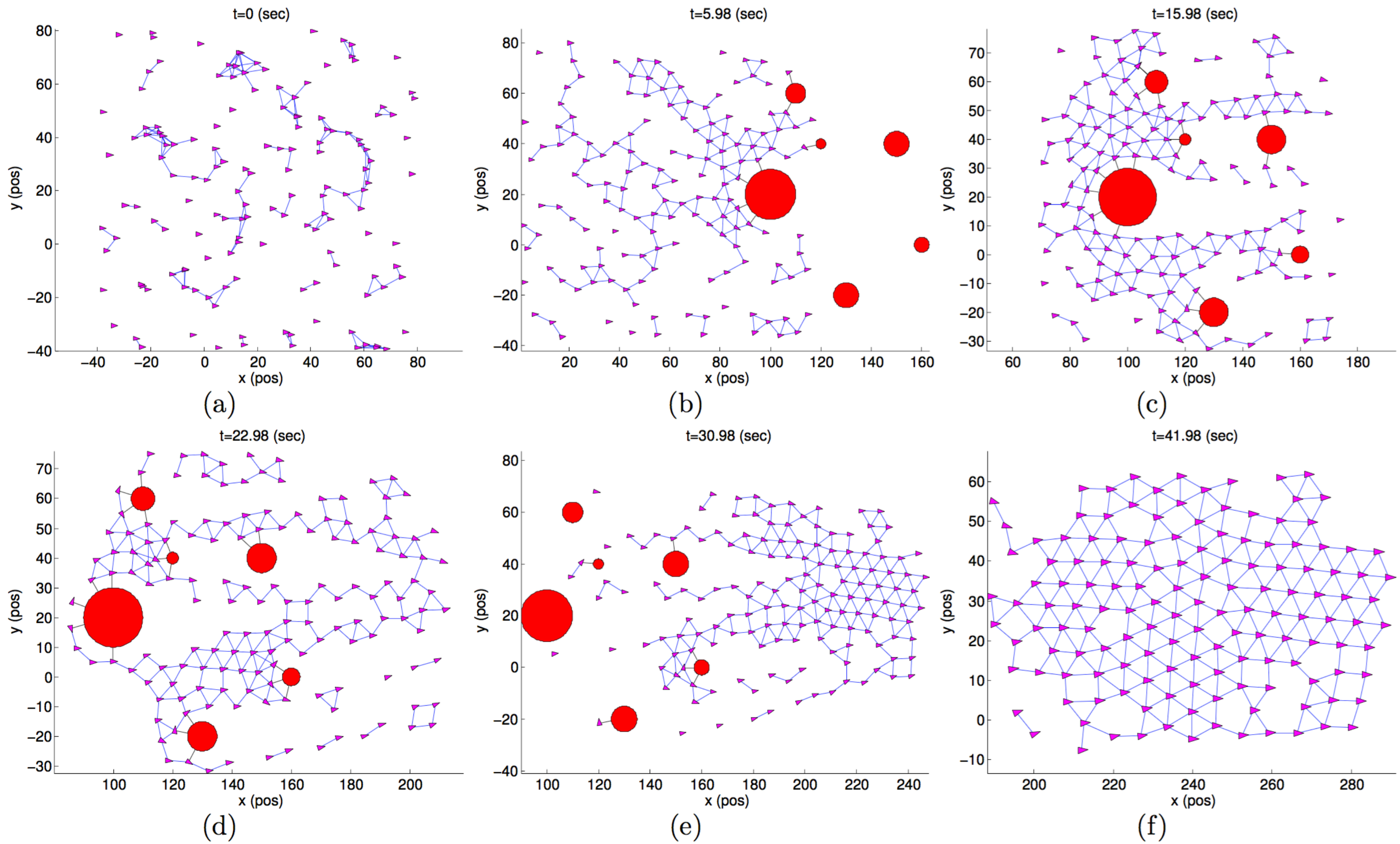


Figure 15: The split/rejoin maneuver for $n = 150$ agents.

Simulation Results

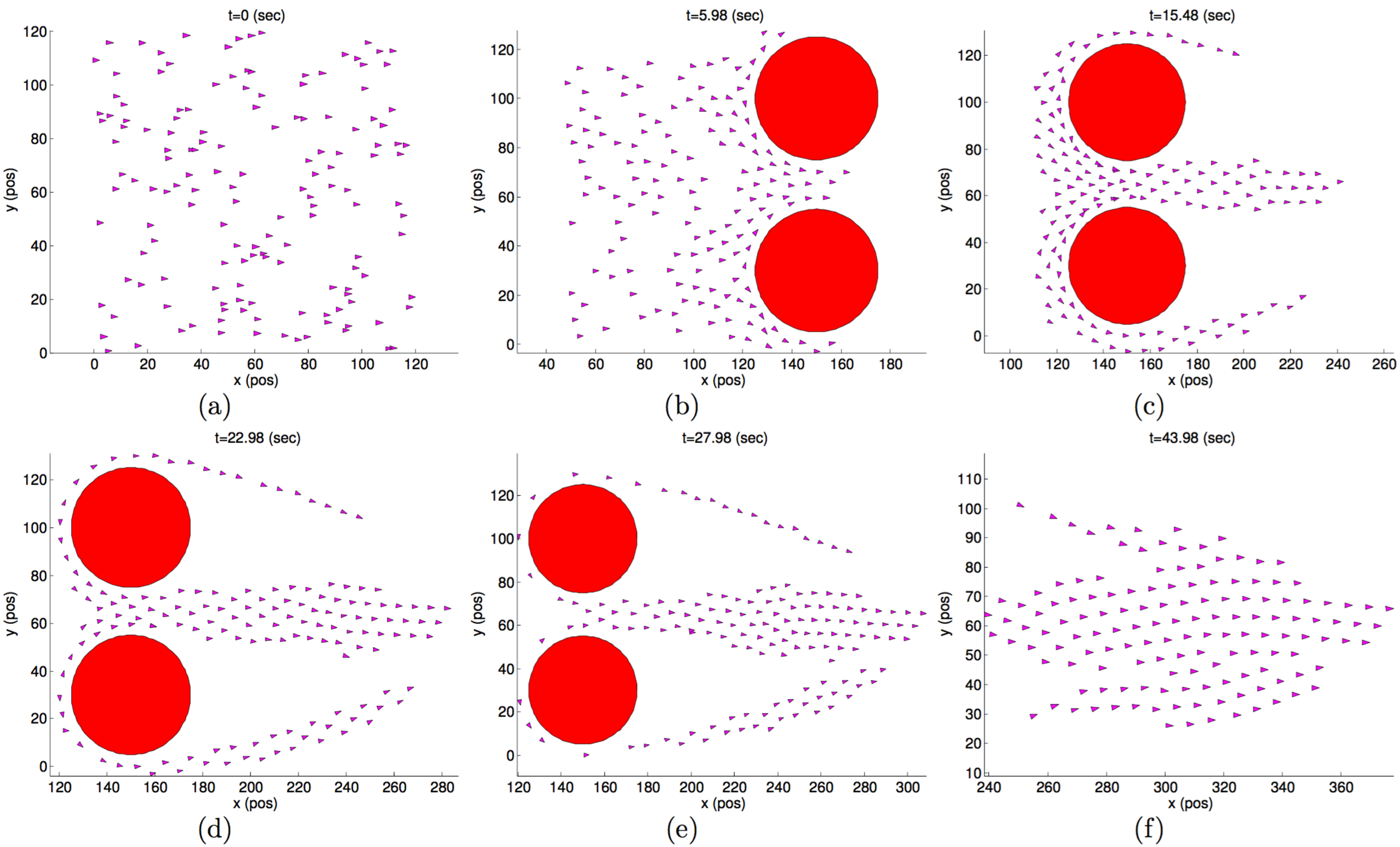


Figure 16: The squeezing maneuver for $n = 150$ agents.

Simulation Results

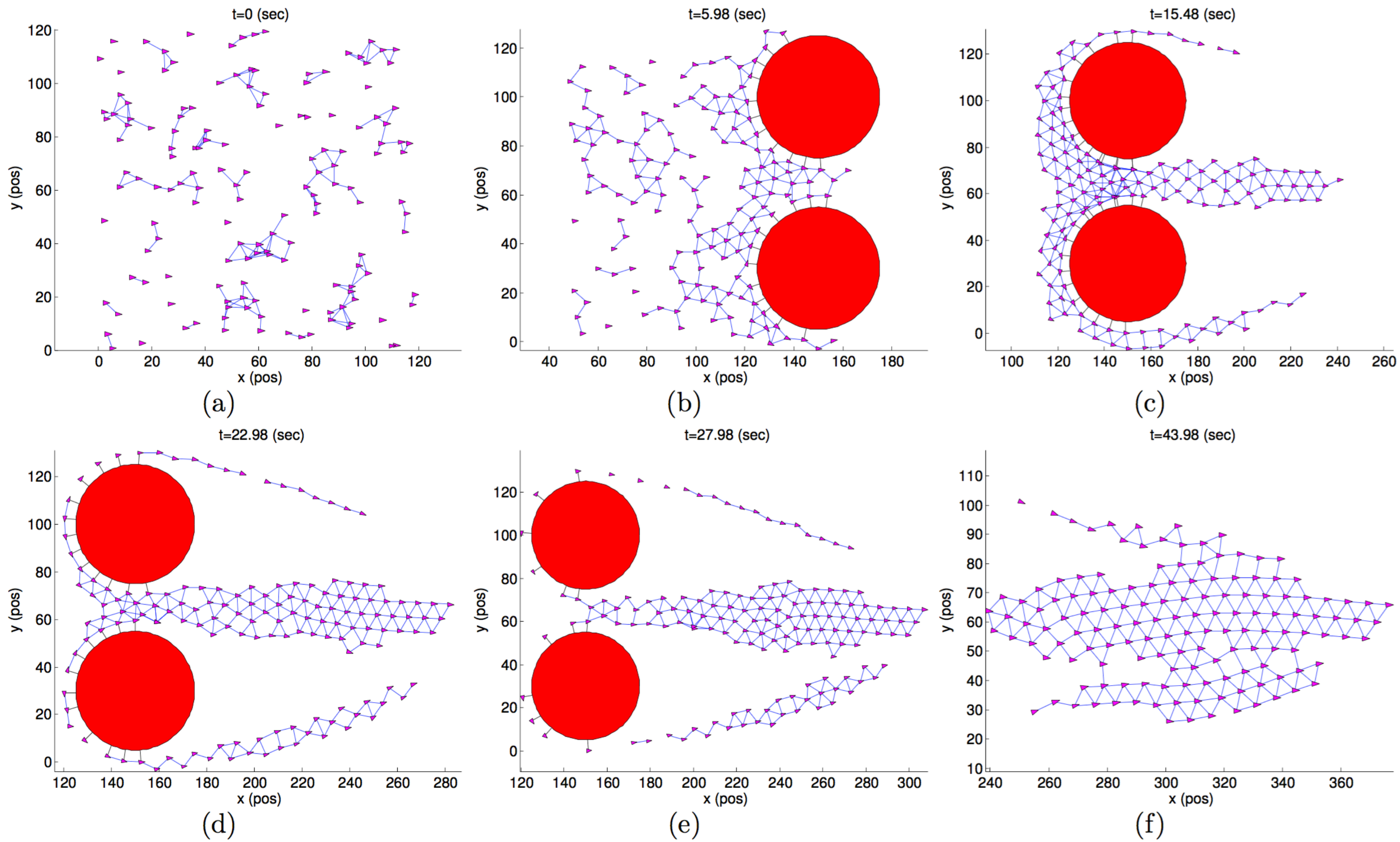


Figure 17: The squeezing maneuver for $n = 150$ agents.

What Constitutes Flocking?

- Define flocking in a way that is objective and independent of algorithm

Definition 5. (α-flocking) Let $z : t \mapsto \text{col}(q(t), p(t))$ be the state trajectory of a system of n dynamic agents (or particles). We say a group of agents perform *α-flocking* over the time interval $[t_0, t_f]$ if there exists relatively small numbers $\epsilon_0, \epsilon_1, \epsilon_2 > 0$ and a distance $d > 0$ such that the trajectory $z(t)$ satisfies all the following conditions for all $t \in [t_0, t_f]$ with an interaction range $r = (1 + \epsilon_0)d$:

i) The group remains a quasi-flock (i.e. $G(q(t))$ has a giant component).

ii) The group remains cohesive.

iii) The deviation energy remains small (or $E(q(t)) \leq \epsilon_1 d^2$).

iv) The velocity mismatch remains small (or $K(v(t)) \leq \epsilon_2 n$).

A more strict form of flocking, or *strict α-flocking*, can be defined by replacing the above four conditions with the following three properties:

a) The group remains a flock (i.e. net $G(q(t))$ is connected).

b) The deviation energy remains small (i.e. $E(q(t)) \leq \epsilon_1 d^2$).

c) The velocity mismatch remains small (i.e. $K(v(t)) \leq \epsilon_2 n$).

What Constitutes Flocking?

- To verify α-flocking, calculate 4 quantities:
  - Relative Connectivity

$$C(t) = \frac{1}{n-1}\operatorname{rank}(\hat{L}(q(t)))$$

  - Cohesion Radius

$$R(t) = \max_{i \in \mathcal{V}} \|q_i(t) - q_c(t)\|$$

  - Normalized Deviation Energy

$$\tilde{E}(q) = E(q)/d^2$$

  - Normalized Velocity Mismatch

$$\tilde{K}(v) = K(v)/n$$

What Constitutes Flocking?

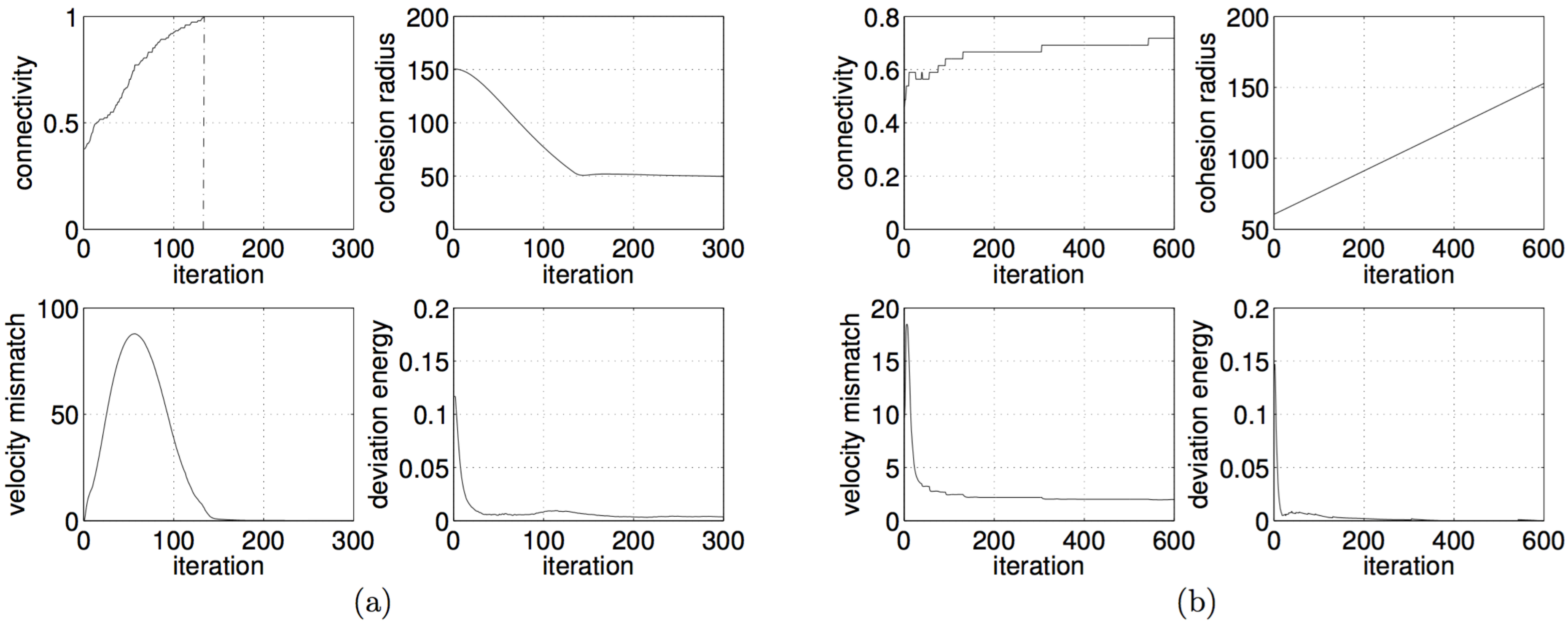


Figure 18: The $C, R, \tilde{E}, \tilde{K}$ curves for simulations in Fig. 11 and Fig. 12: (a) flocking and (b) regular fragmentation.

What Constitutes Flocking?

- More accurate measure of connectivity:

$$C^* = \frac{|\mathcal{E}(\Gamma^*)|}{|\mathcal{E}(G)|}$$

- 5 components with population 1,1,1,1,96:
  - C = 95/99
  - C* = 0.96
- 5 components with population 20,20,20,20,20:
  - C = 95/99
  - C* = 0.2

Conclusions

- The theoretical framework presented is versatile and has desirable proven properties
- Used to build 3 flocking algorithms
- Simulations demonstrate the theoretical results of the framework
- Presents an algorithm-independent definition of flocking

References

- Olfati-Saber, R. (2006). Flocking for multi-agent dynamic systems: algorithms and theory, IEEE Transactions on Automatic Control, 51(3):401-420.