Bio-inspired Collective Behavior of Autonomous Outdoor Drone Swarms

Gábor Vásárhelyi, PhD
Eötvös University, Budapest, Hungary
Department of Biological Physics
MTA-ELTE Statistical and Biological Physics Research Group, Budapest, Hungary
CollMot Robotics Ltd.

IEEE RAS Summer School on Multi-Robot Systems
Prague, Czech Republic, 2019.07.29.
Eötvös University, Department of Biological Physics research group of Prof. Tamás Vicsek, Budapest, Hungary
Film by Andrei Golovnev / www.facebook.com/andrei.golovnev.9
Key descriptors of collective motion

- **Non-equilibrium** complex system
- **Universal** rules and patterns
- **Local** decisions $\rightarrow$ **global** behavior
- **Simple** rules $\rightarrow$ **complex** dynamics
- Local decisions and **interactions** rule the world
- **Decentralized**, autonomous, robust, redundant
- Fully **optimized** through evolution and learning
- **Emergent** complexity, swarm is more than the sum of individuals
Stable hierarchy in the collective flight of 30 homing pigeons

Multi-level hierarchy in wild horses

Collective thermalling of storks

Nature → Collective Robotics

Collective control

Optimization algorithms

Structure of complexity
Lets create a bio-inspired flock of drones, that is…

- **Autonomous, decentralized** (all HW+SW, -GPS)
- **Scalable in size and speed** (1000→1 vs 1→1000)
- Works **outdoor** → flocking algorithms should tolerate noise + delay
- Fleet is controlled as a **meta unit** and performs tasks in a self-organized manner
- Based on stable local problem solving and decision making fleet can be used for **practical collaborative applications**
Largest “swarms”: show business

Intel’s record-breaking drone show
What is needed for our autonomous drone flock?

1) extendible drones
2) real-time computing
3) sensors
4) communication

HW
What is needed for an autonomous drone flock?

1) extendible drones
2) real-time computing
3) sensors
4) communication
5) realistic simulation framework

HW

SW
What is needed for an autonomous drone flock?

1) extendible drones
2) real-time computing
3) sensors
4) communication

5) realistic simulation framework
6) flocking/swarming algorithms

7) complex error handling
AutoPilot: Pixhawk
Companion Computer: Odroid / Raspberry Pi
Communication: XBee / 900 MHz SiK Telemetry + Ad Hoc Wifi
Positioning: GNSS

Environment: Linux, C / Python
Low-level control: ArduCopter + MAVLink
High-level control: FlockCtrl (proprietary)
Realistic simulation framework

- Agent-based modeling optimized for **flock-level** thinking
- **GUI** for real time visualization and parameter tuning
- Feasible execution times for up to ~1000 agents
- Optimized for **parallel computing**
- **Evolutionary optimization** module

Code available at [https://github.com/csviragh/robotsim](https://github.com/csviragh/robotsim)
Why is it “realistic”? 

- Realistic (but simplified) **dynamic model** of agents:
  - Inertia and limited acceleration
  - Limited refresh rate of sensors
  - Delayed response

- Detailed modeling of internal and external **disturbances**:
  - Input noise: realistic GNSS noise model
  - Output noise: Gaussian noise on acceleration
  - Communication limits, delays, errors

Code available at [https://github.com/csviragh/robotsim](https://github.com/csviragh/robotsim)
Problem statement

Realistic model with differential velocity updates:

\[
\dot{v}_i(t + \Delta t) = \dot{v}_i(t) + \frac{1}{\tau} \left( v_i^{\text{desired}}(\Delta \ddot{x}_{ij}(t^-), \Delta \dot{v}_{ij}(t^-), ...) - \dot{v}_i(t) \right) \Delta t + \mu(t) \Delta t
\]

Define \( v_i^{\text{desired}} \) so that

Pairwise interactions based on momentary states of local agents (flocking) …

are mixed optimally with

… information about future using prediction of motion of others and self (motion planning).
Flocking algorithm essentials

**Long range attraction:** units too far approach the flock

**Medium range alignment:** units around equilibrium distance try to align their velocities

**Short range repulsion:** units too close push away each other

**Enemies:** delay, noise, inertia, finite acceleration, communication errors
Desired velocity

\[ \mathbf{\dot{v}}_i^{\text{desired}} = \sum_{j \neq i} \mathbf{\dot{v}}_{ij}^{\text{repulsion}} + \sum_{j \neq i} \mathbf{\dot{v}}_{ij}^{\text{friction}} + \sum_{o} \mathbf{\dot{v}}_{io}^{\text{hill}} + \mathbf{\dot{v}}_i^{\text{task}} \]

\[ \mathbf{\dot{v}}_{i}^{\text{repulsion}} = \begin{cases} -D \sum_{j \neq i} \left( |\mathbf{x}_j - \mathbf{x}_i| - r_0 \right) \frac{\mathbf{\dot{x}}_j - \mathbf{\dot{x}}_i}{|\mathbf{x}_j - \mathbf{x}_i|}, & \text{if } |\mathbf{x}_j - \mathbf{x}_i| < r_0 \\ 0, & \text{otherwise, repulsion (induces oscillations)} \end{cases} \]

\[ \mathbf{\dot{v}}_{i}^{\text{friction}} = C \sum_{j \neq i} \frac{\mathbf{\dot{v}}_j - \mathbf{\dot{v}}_i}{(\mathbf{x}_j - \mathbf{x}_i)^2}, \quad \text{alignment (reduces oscillations)} \]

\[ \mathbf{\dot{v}}_{i}^{\text{task}} = \begin{cases} v_0 \frac{\mathbf{\dot{v}}_{i}^{\text{COM}} + \mathbf{\dot{v}}_{i}^{\text{form}}}{|\mathbf{\dot{v}}_{i}^{\text{COM}} + \mathbf{\dot{v}}_{i}^{\text{form}}|}, & \text{if } |\mathbf{\dot{v}}_{i}^{\text{COM}} + \mathbf{\dot{v}}_{i}^{\text{form}}| > v_0 \\ \mathbf{\dot{v}}_{i}^{\text{COM}} + \mathbf{\dot{v}}_{i}^{\text{form}}, & \text{otherwise, spp/drive, e.g., formation + target tracking} \end{cases} \]
Slow information spreading
SHOCKWAVE TRAFFIC JAMS RECREATED FOR FIRST TIME

Footage courtesy of University of Nagoya, Nagoya, Japan
<table>
<thead>
<tr>
<th>Key</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>v</td>
<td>On</td>
</tr>
<tr>
<td>a</td>
<td>On</td>
</tr>
<tr>
<td>q</td>
<td>2.0</td>
</tr>
<tr>
<td>l</td>
<td>8.00</td>
</tr>
<tr>
<td>2</td>
<td>2.0</td>
</tr>
<tr>
<td>3</td>
<td>0.0</td>
</tr>
<tr>
<td>4</td>
<td>6.5</td>
</tr>
<tr>
<td>5</td>
<td>2.00</td>
</tr>
<tr>
<td>6</td>
<td>2.00</td>
</tr>
<tr>
<td>7</td>
<td>1.50</td>
</tr>
<tr>
<td>8</td>
<td>0.00</td>
</tr>
<tr>
<td>9</td>
<td>0.000</td>
</tr>
<tr>
<td>b</td>
<td>1.00</td>
</tr>
<tr>
<td>c</td>
<td>1.00</td>
</tr>
<tr>
<td>d</td>
<td>14.6</td>
</tr>
<tr>
<td>r</td>
<td>0.20</td>
</tr>
<tr>
<td>j</td>
<td>100.00</td>
</tr>
<tr>
<td>s</td>
<td>50</td>
</tr>
</tbody>
</table>

Elapsed time: 0.0 sec
Collective “follow-me”

G. Vásárhelyi, Cs. Virágh, N. Tarcai, T. Szörényi, G. Somorjai, T. Nepusz and T. Vicsek,
„Outdoor flocking and formation flight with autonomous aerial robots,” IROS 2014
Formation flights

Flocking in a confined area (2014)

Expectations towards friction/alignment

• Should be a local interaction
• Should reduce pairwise velocity difference
• Should not dominate repulsion
• Should reduce oscillations induced by repulsion and/or task specific terms
• Should work for large velocity domains
• Algo specific requirements/goals:
  – **flocking**: local + global alignment
  – **traffic**: selective local alignment + global independence
Braking curve and stopping distance

- **Braking curve**
  - Various curves for different acceleration values and times.
  - Legend includes:
    - `const a=2.5 m/s^2`
    - `linsqrt a=2.5 m/s^2, p=0.5 1/s`
    - `linsqrt a=2.5 m/s^2, p=1.0 1/s`
    - `const a=5.0 m/s^2`
    - `linsqrt a=5.0 m/s^2, p=0.5 1/s`
    - `linsqrt a=5.0 m/s^2, p=1.0 1/s`

- **Stopping distance**
  - Various stopping distances for different velocities and accelerations.
  - Legend includes:
    - `a=2.5 m/s^2, p=0.5 1/s, v'=5.00 m/s`
    - `a=2.5 m/s^2, p=1.0 1/s, v'=2.50 m/s`
    - `a=5.0 m/s^2, p=0.5 1/s, v'=10.00 m/s`
    - `a=5.0 m/s^2, p=1.0 1/s, v'=5.00 m/s`
Desired velocity

\[ \vec{v}_i^{\text{desired}} = \sum_{j \neq i} \vec{v}_ij^{\text{repulsion}} + \sum_{j \neq i} \vec{v}_ij^{\text{friction}} + \sum_{o} \vec{v}_io^{\text{shill}} + \vec{v}_i^{\text{task}} \]
Optimization (model tuning) is necessary

→ Evolution

- Stochastic optimization, no guarantee of global optimum!
- **Efficiency**: In real models there are many parameters (>5). Parameter sweeping gets slow, manual tuning fails
- **Universality**: “Evolution is the second-best optimization method for all problems”
- **Bio-inspiration**: evolution is the optimization principle of nature itself. It must work if you can read this!
- **Simplicity**: All you need is a good fitness function…
Fitness function (for flocking)

- Bring average velocity close to flocking speed
- Create large flocking connected clusters
- Maximize velocity correlation in connected clusters
- Avoid collisions with each other
- Avoid collisions with arena walls and obstacles
Flocking in confined areas (2017)

6 m/s, collective object avoidance

8 m/s, emergent circular flight

Flocking vs traffic
How to do this with UAVs in 3D?
Two-point package delivery

High density traffic in smart cities

Decentralized airspace coordination

Traffic interaction terms

• **Anisotropic repulsion**: avoid frontal collisions and slip around each other

• **Selective alignment**: no need to align to e.g. those that are behind and go elsewhere

• **Agile self-drive**: prevent collisions by avoiding risky situations in time, reduce speed if needed, be patient

• **Radial queueing**: wait for common target patiently in queue, do not over-excite system
Optimized drone traffic (2018)

6 m/s, crosswalk traffic of drones 6 m/s queueing simulation

Collective chase and escape

Financial Support

• EU ERC COLLMOT (2009-2015)
• Zoltán Magyary Postdoctoral Fellowship, TÁMOP 4.2.4.A/1-11-1-2012-0001 (2013-2014)
• Bolyai Research Scholarship of the Hungarian Academy of Sciences BO/00219/15/6 (2016-2018)
• USAF Research Grant No. FA9550-17-1-0037 (2017-2018)
• MTA-ELTE Statistical and Biological Physics Research Group (2012-2017-2022)
• K_16 Research Grant of the Hungarian National Research, Development and Innovation Office, K 119467 (2016-2020)
• CollMot Robotics Ltd. (2015-)
Thanks to the team

Drone related publications


• B. Boldizsár, G. Vásárhelyi, “Coordinated dense aerial traffic with self-driving drones,” *ICRA 2018*, pp 6365–6372


• G. Vásárhelyi, Cs. Virágh, N. Tarcai, T. Szörényi, G. Somorjai, T. Nepusz and T. Vicsek, *Outdoor flocking and formation flight with autonomous aerial robots*, *IROS 2014*


hal.elte.hu/~vasarhelyi  |  vasarhelyi@hal.elte.hu