

EvoFlock: evolved inverse design of multi-agent motion

Optimizing multi-agent motion models with
procedural objectives and genetic algorithms

these slides:



Craig Reynolds — Artificial Life — Waterloo, Ontario, Canada — August 20, 2026

Inspired by Susan Stepney's framing,
this is an “engineering” talk:
make an ALIFE model do what we want.

Multi-agent motion models are used to represent:

bird flocks
mammal herds
fish schools
human crowds
robotic swarms
scientific simulations
...

“flocks”

They can be difficult to work with

The goal is to control group behavior but
the parameters control individuals.

Parameters interact and
have non-linear effect.

Changing one usually requires
changing other to compensate.

**These problems can be
addressed with *inverse design***

We can tune flocks using an objective function and optimization

Create a objective function
to “score” a flock simulation.

Optimize parameters toward the
design goal expressed in objective.

Why tune or adjust flocks?

Biological simulation: birds versus fish.
Animation storytelling: happy versus sad.

Collaborators

University of Washington

Gilbert Bernstein

Dianlun (Jennifer) Luo

Ryan Zambrotta

Matthew Shang

University of California San Diego

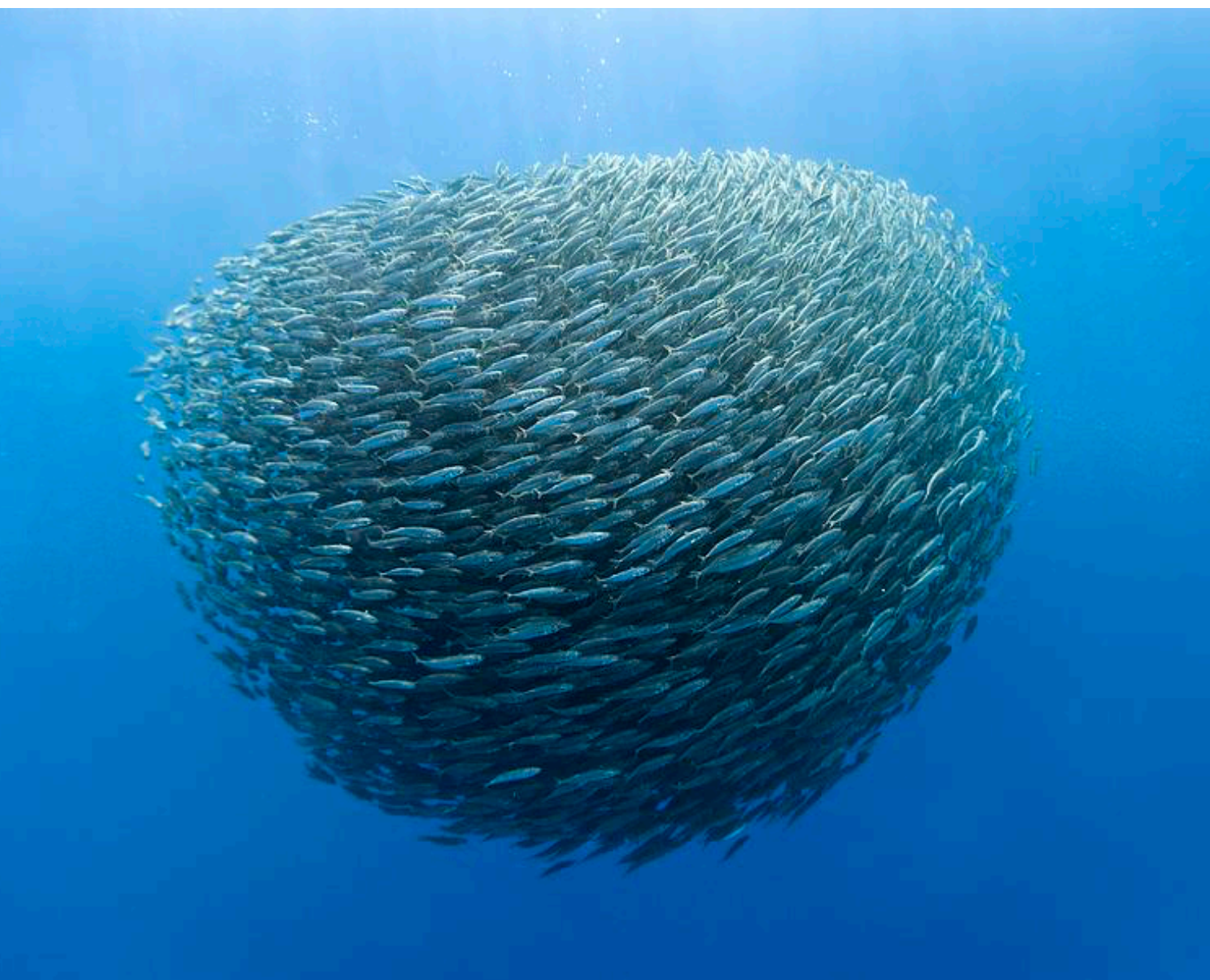
Tzu-Mao Li

My teammates are building a differentiable simulation
to allow using gradient descent.

In parallel I used gradient-free evolutionary optimization.

Early boids
1986-1987

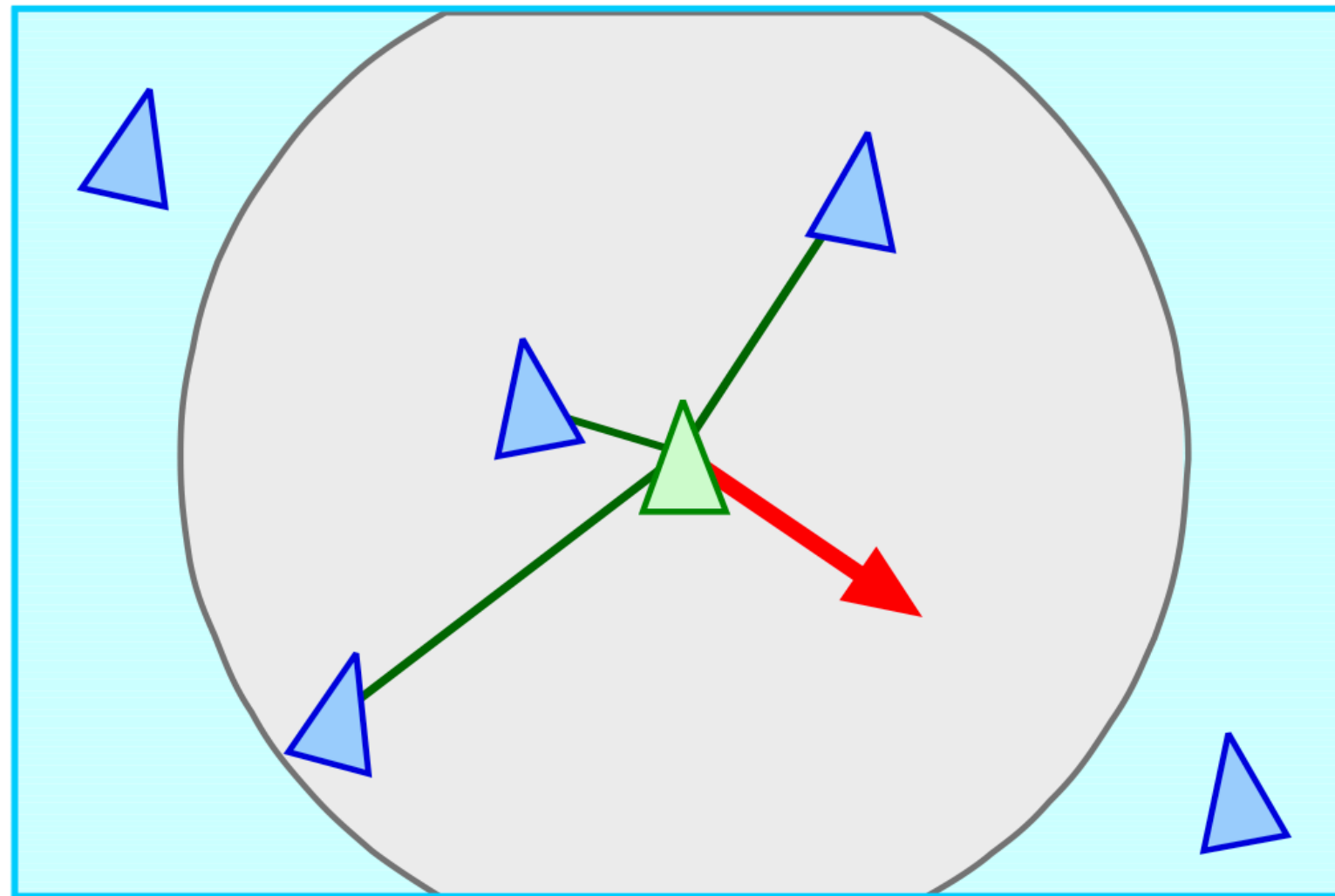
A childhood fascination with natural complexity



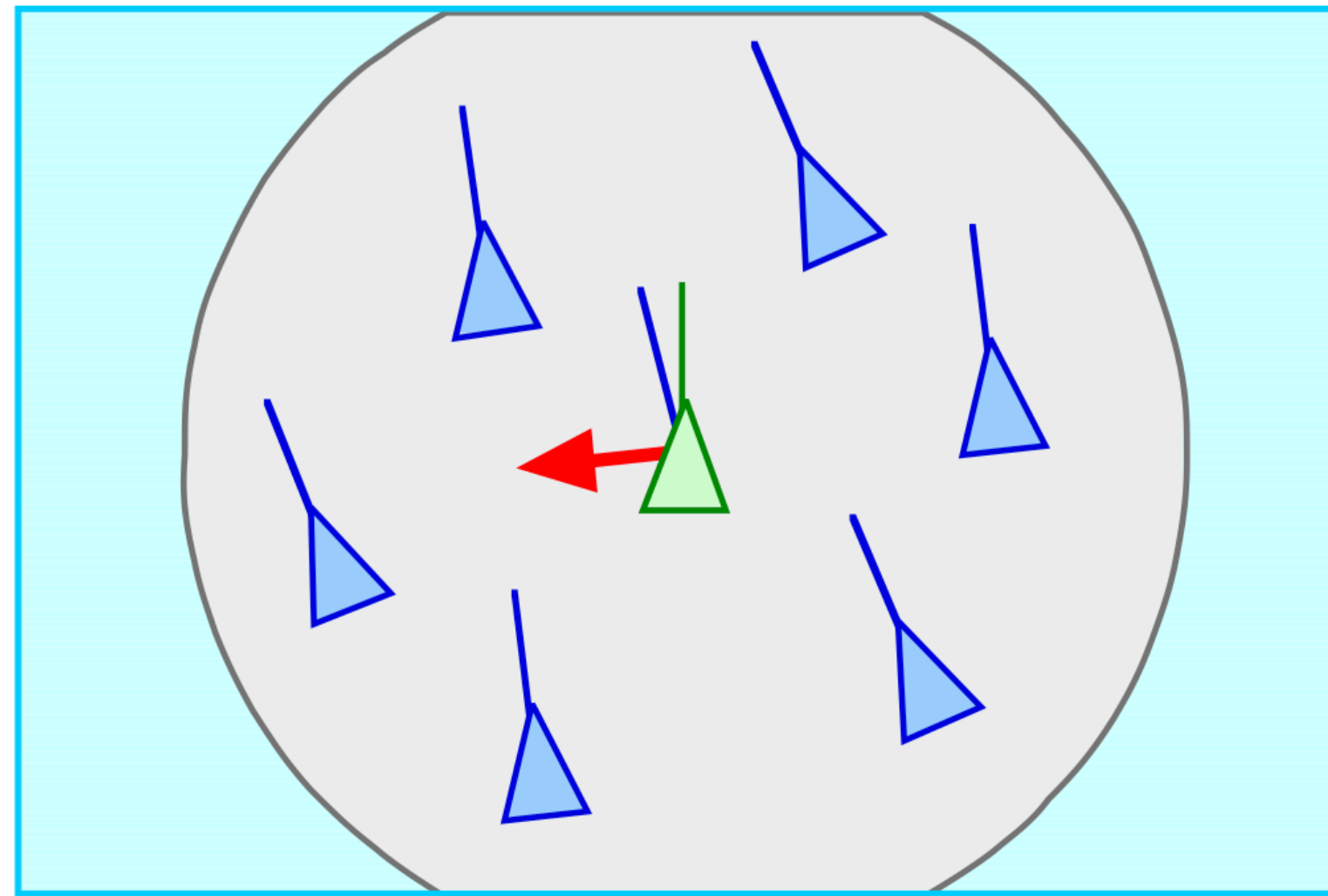


Three components of flocking

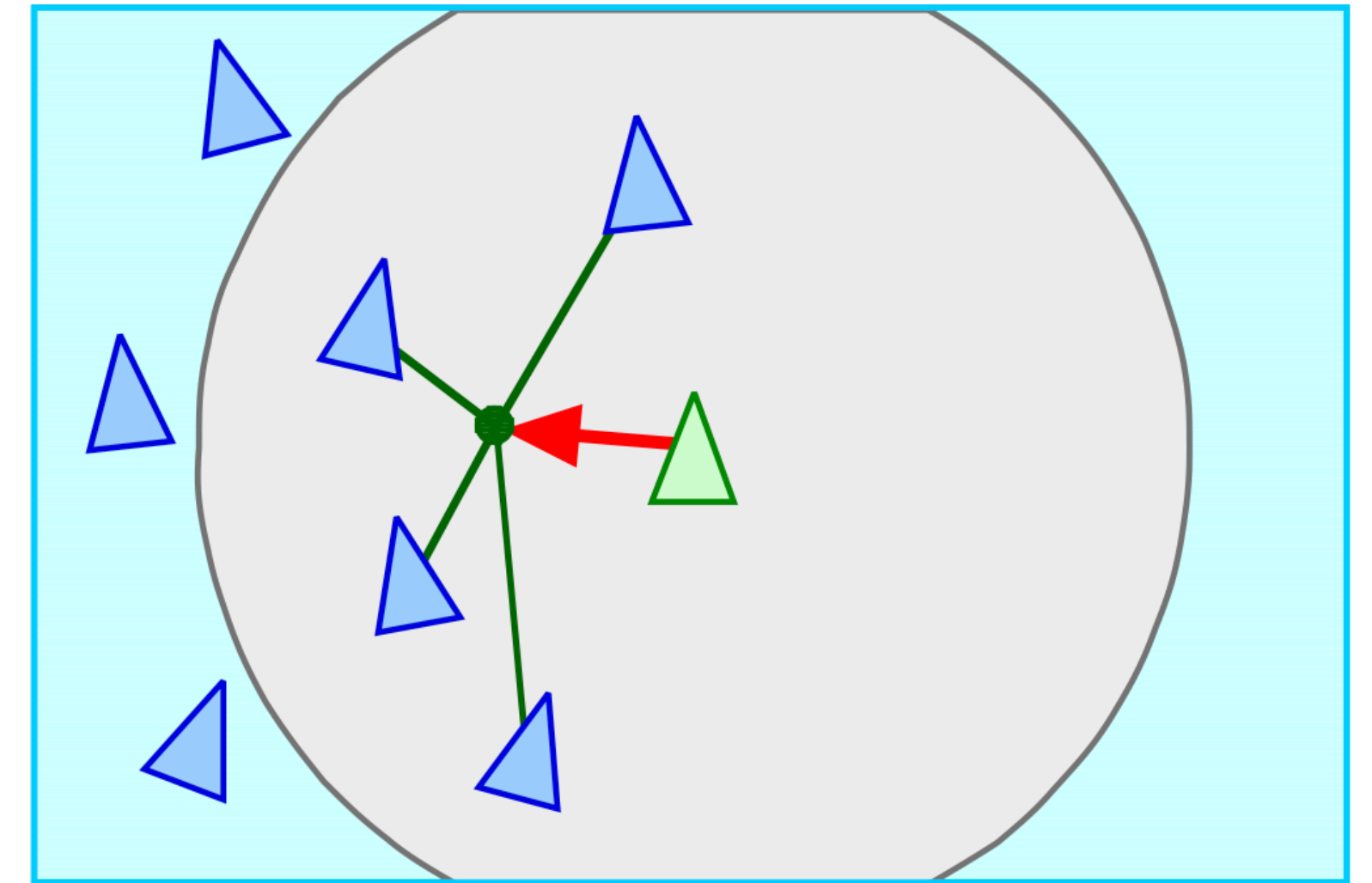
Seemed necessary, but were they sufficient?



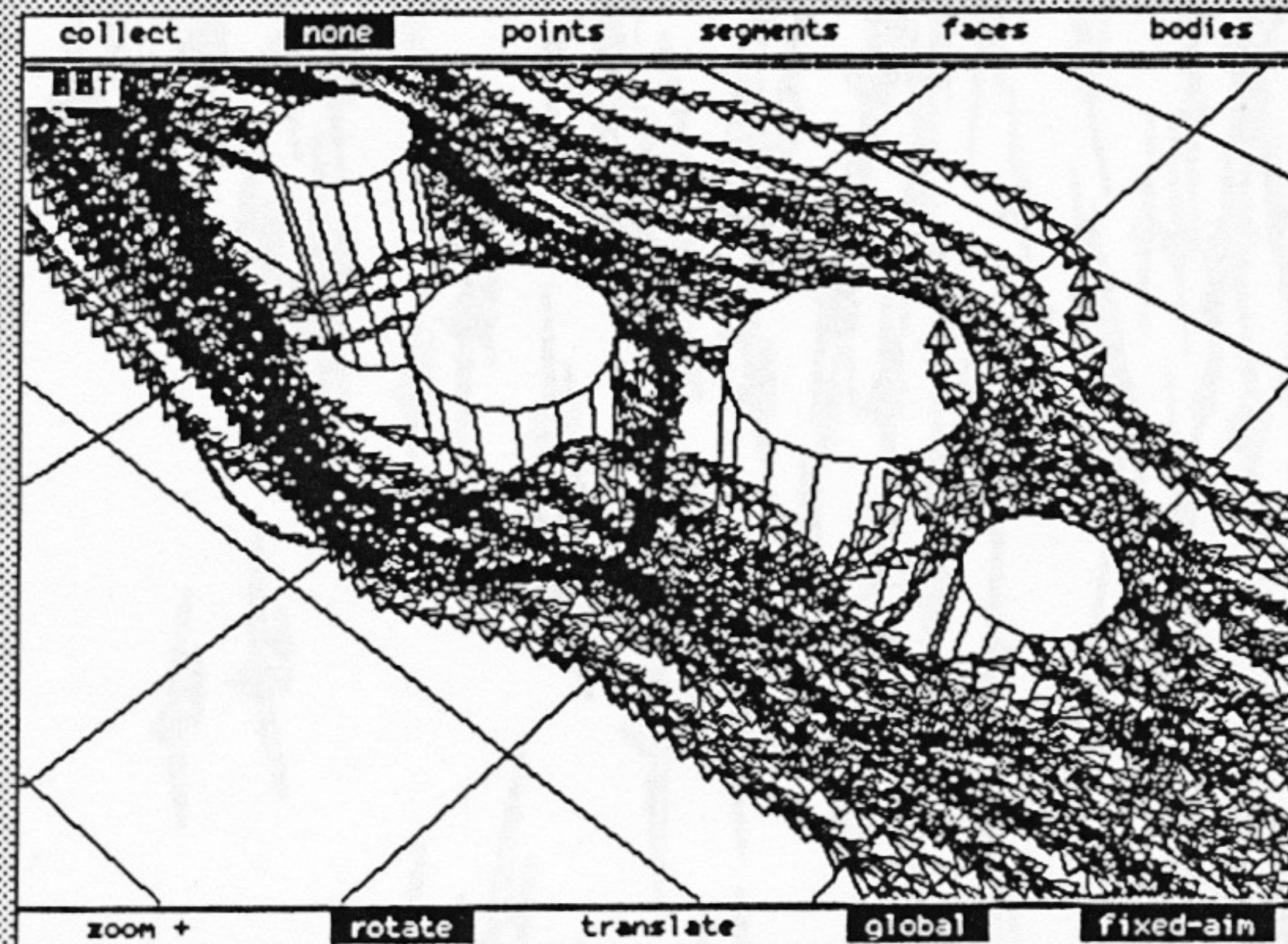
Separation



Alignment



Cohesion



S-GEOMETRY / DYNAMICS

Camera:

Parameters
Modes
.Views.
.Align.
Aim
Level

Object Display:

.Visibility.
Backfacing
Silhouetting
Hard Edging
Points
Substitute Boxes

Utilities:

Rename
Setq J
.Print Data.
Add Attachment
.Number.
.Kill.

Output:

File
.Hardcopy.
.Render.
.Shade.
Paint

Command: (setq *playback-alu* tv:alu-for)

7

Command: █

3d Live Window 1

S-Dynamics Script Editor

Script "Flocking-Around" (30 fps, 2 sec)

Fly flock

Camera

side camera

Scenery

0.000

absolute time (seconds)

2.000

2.000

60

Select sequence

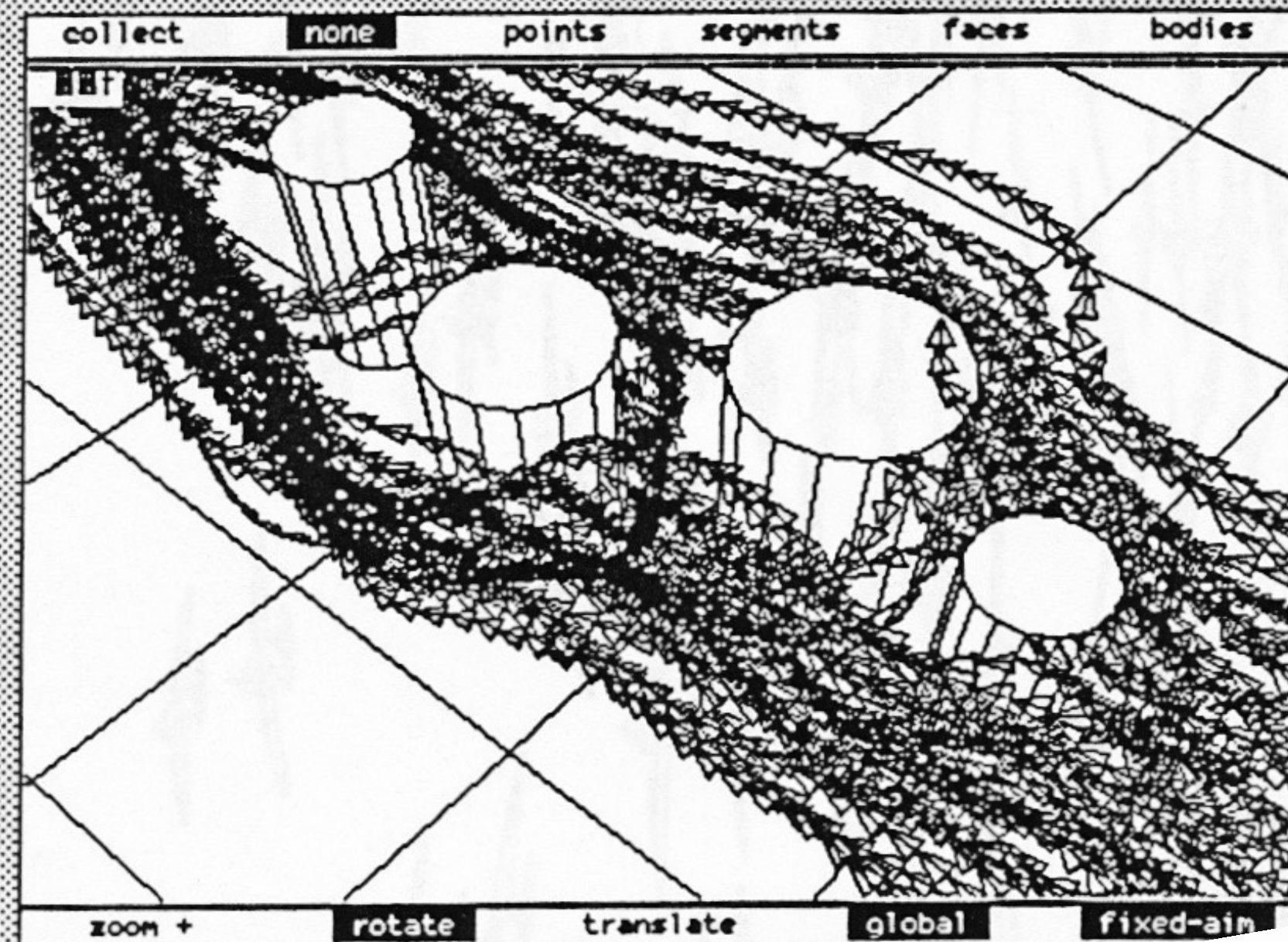
Save script

Display frame

Animate script

Playback

Change modes



S-GEOMETRY / DYNAMICS

Camera:

Parameters
Modes
.Views.
.Align.
Aim
Level

Object Display:

.Visibility.
Backfacing
Silhouetting
Hard Edging
Points
Substitute Boxes

Utilities:

Rename
Setq J
.Print Data.
Add Attachment
.Number.
.Kill.

Output:

File
.Hardcopy.
.Render.
.Shade.
Paint

Command: (setq *playback-alu* tv:alu-for)

7

Command: █

S-Dynamics Script Editor

```
(defmethod (:steer flight-mixin) (old-velocity new-velocity))
(vlet* ((new-adjusted-velocity (send self :constrain-steering new-velocity))
      (new-adj-acceleration (3d-vector-sub new-adjusted-velocity old-velocity))
      (actual-steering-acceleration (truncate-magnitude new-adj-acceleration
                                                         (send self :max-acceleration))))
  (actual-new-velocity (3d-vector-add old-velocity actual-steering-acceleration))
  (when (plusp (magnitude-squared actual-new-velocity))
    ;; PITCH and YAW the object (local X and Y rotations) to align with the new local velocity.
    (send self :align-to-local actual-new-velocity)
    ;; Adjust the roll for a "coordinated turn".
    (send self :bank actual-steering-acceleration))))
```

Scenery

0.000

absolute time (seconds)

2.000

2.000

60

Select sequence

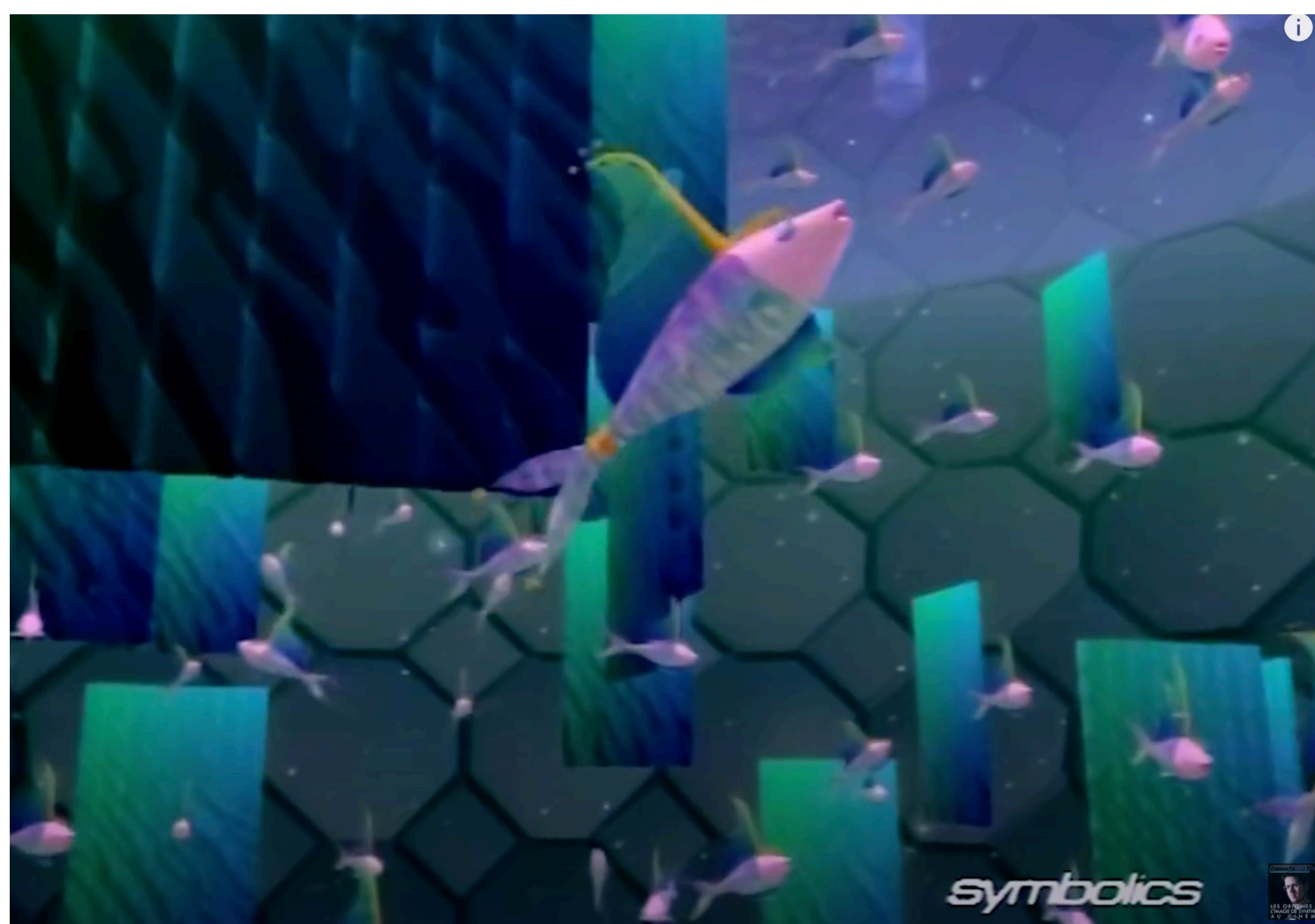
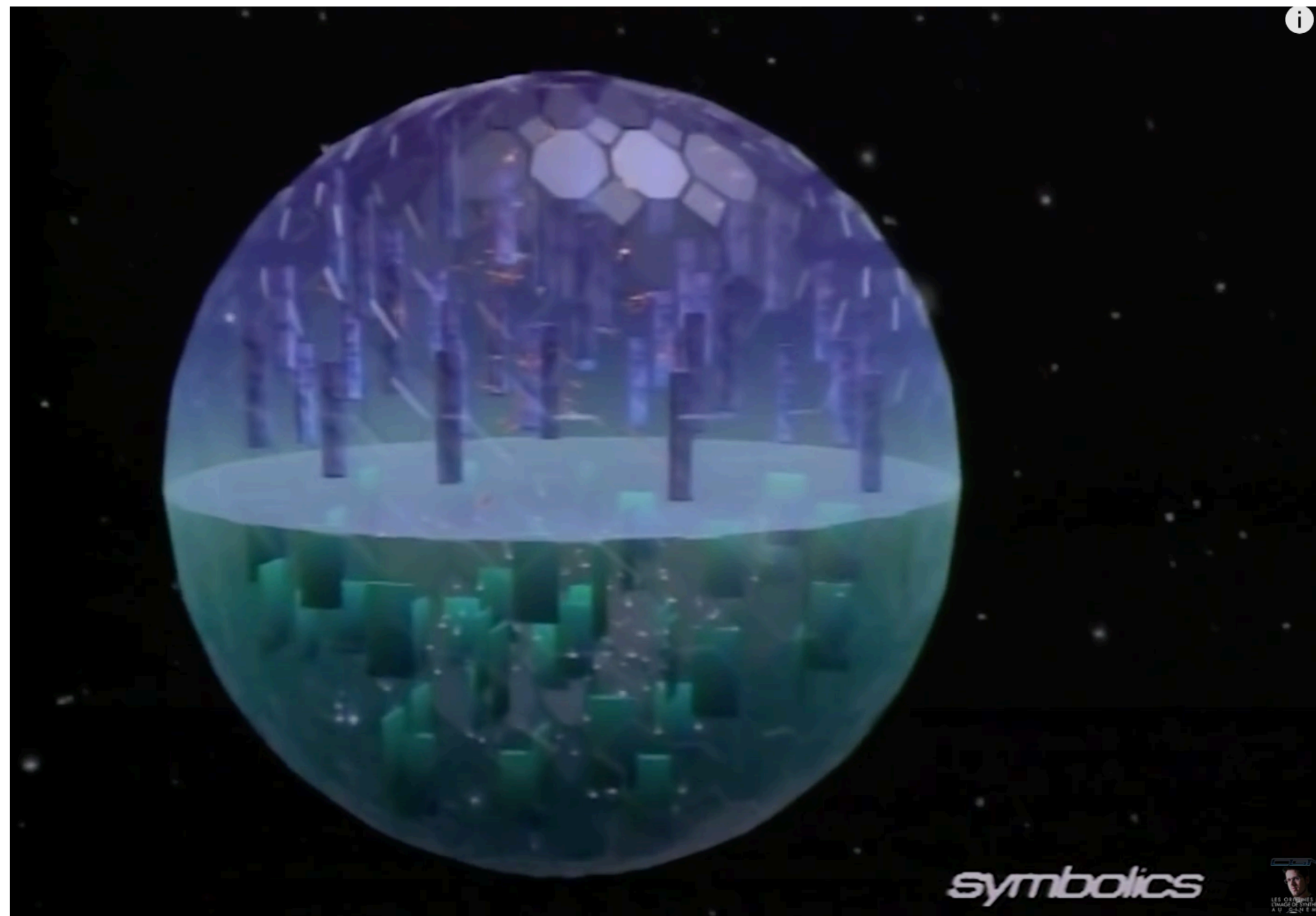
Save script

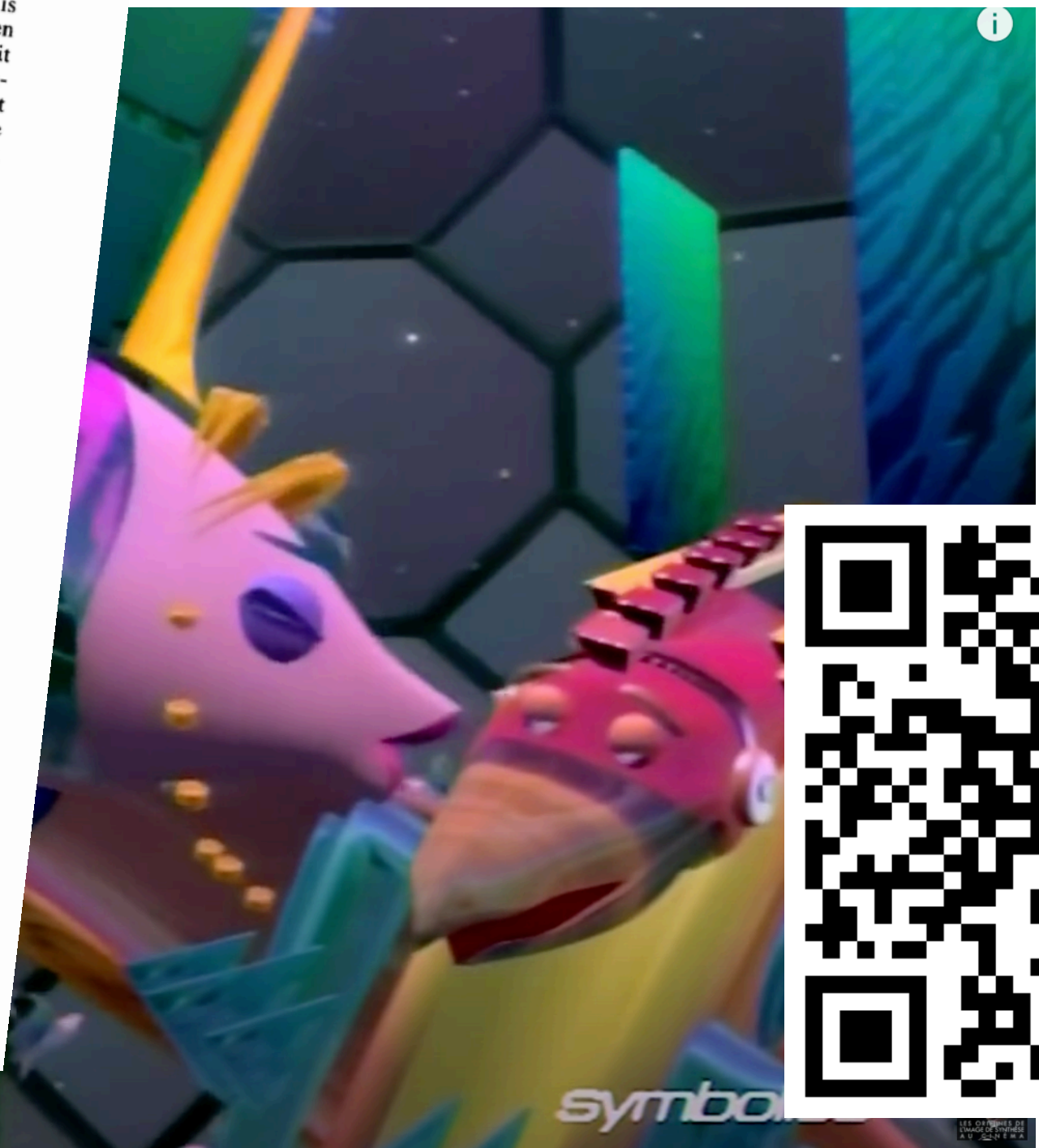
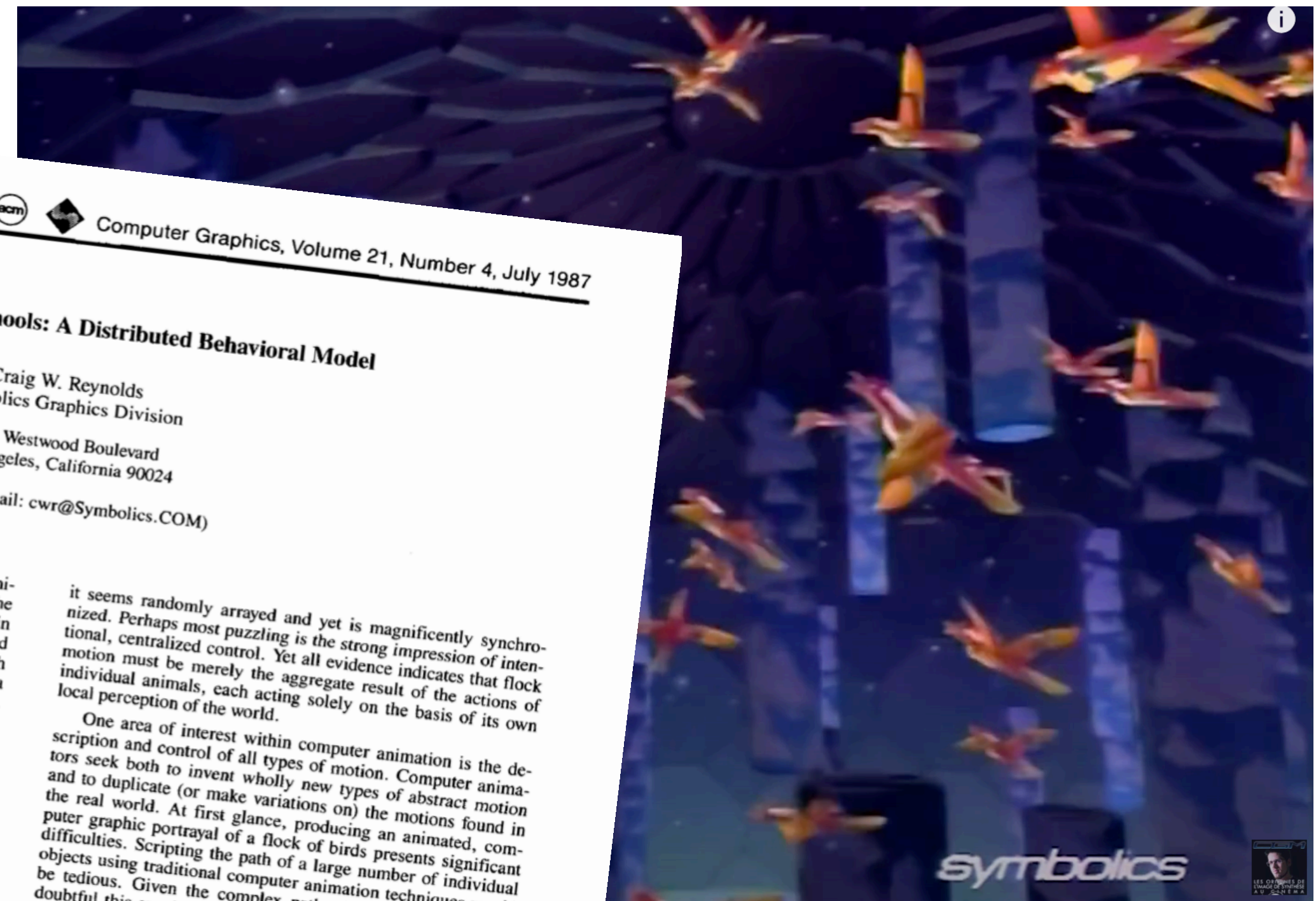
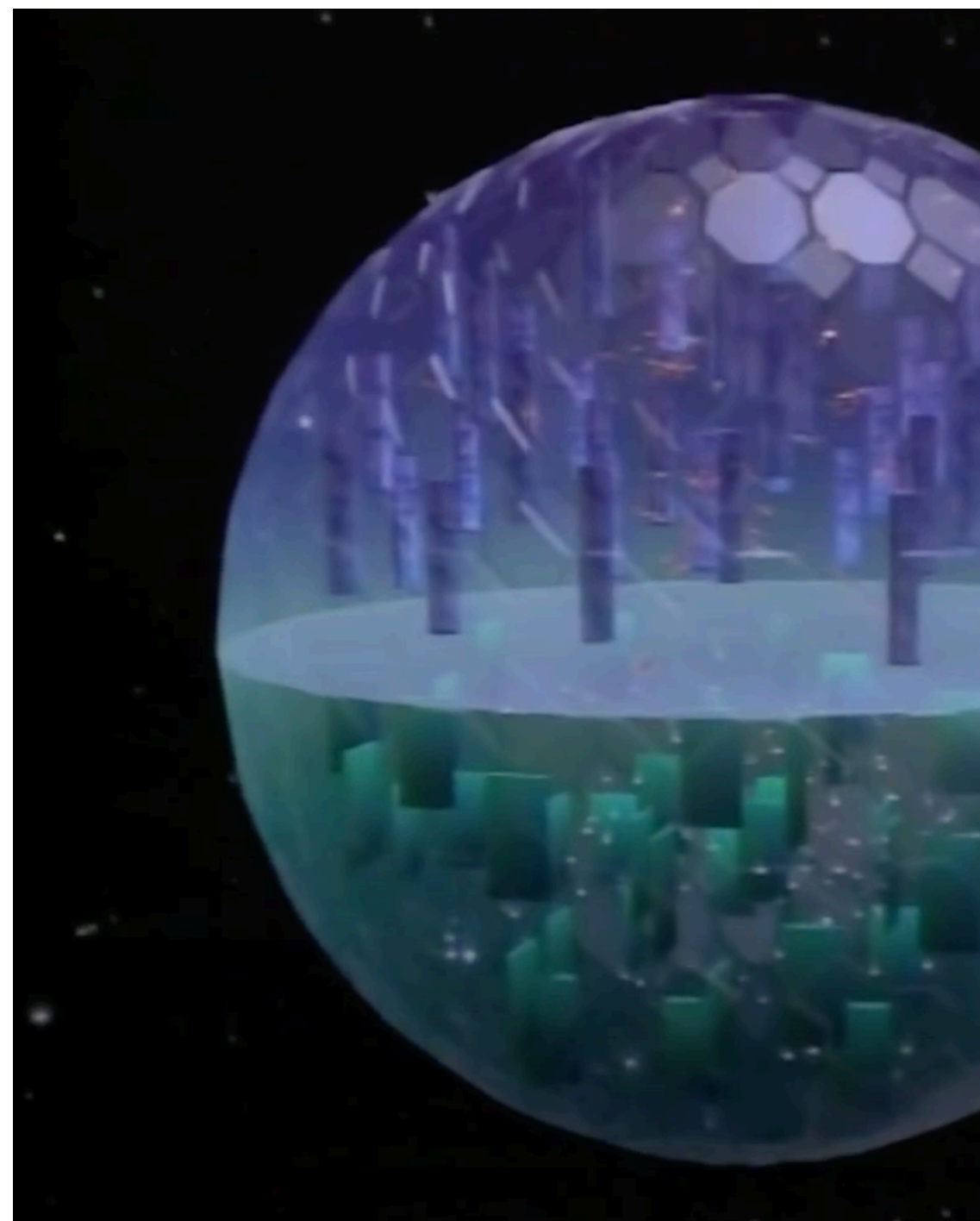
Display frame

Animate script

Playback

Change modes







Computer Graphics, Volume 21, Number 4, July 1987

Flocks, Herds, and Schools: A Distributed Behavioral Model

Craig W. Reynolds
Symbolics Graphics Division
1401 Westwood Boulevard
Los Angeles, California 90024
(Electronic mail: cwr@Symbolics.COM)

Abstract

The aggregate motion of a flock of birds, a herd of land animals, or a school of fish is a beautiful and familiar part of the natural world. But this type of complex motion is rarely seen in computer animation. This paper explores an approach based on simulation as an alternative to scripting the paths of each bird individually. The simulated flock is an elaboration of a particle system, with the simulated birds being the particles. The aggregate motion of the simulated flock is created by a distributed behavioral model much like that at work in a natural flock; the birds choose their own course. Each simulated bird is implemented as an independent actor that navigates according to its local perception of the dynamic environment, the laws of simulated physics that rule its motion, and a set of behaviors programmed into it by the "animator." The aggregate motion of the simulated flock is the result of the dense interaction of the relatively simple behaviors of the individual simulated birds.

Categories and Subject Descriptors: I.2.10 [Artificial Intelligence]: Vision and Scene Understanding; I.3.5 [Computer Graphics]: Computational Geometry and Object Modeling; I.3.7 [Computer Graphics]: Three-Dimensional Graphics and Realism—Animation; I.6.3 [Simulation and Modeling]: Applications.

General Terms: Algorithms, design.

Additional Key Words, and Phrases: flock, herd, school, bird, fish, aggregate motion, particle system, actor, flight, behavioral animation, constraints, path planning.

Introduction

The motion of a flock of birds is one of nature's delights. Flocks and related synchronized group behaviors such as schools of fish or herds of land animals are both beautiful to watch and intriguing to contemplate. A flock* exhibits many contrasts. It is made up of discrete birds yet overall motion seems fluid; it is simple in concept yet is so visually complex,

Permission to copy without fee all or part of this material is granted provided that the copies are not made or distributed for direct commercial advantage, the ACM copyright notice and the title of the publication and its date appear, and notice is given that copying is by permission of the Association for Computing Machinery. To copy otherwise, or to republish, requires a fee and/or specific permission.

© 1987 ACM-0-89791-227-6/87/007/0025 \$00.75

it seems randomly arrayed and yet is magnificently synchronized. Perhaps most puzzling is the strong impression of intentional, centralized control. Yet all evidence indicates that flock motion must be merely the aggregate result of the actions of individual animals, each acting solely on the basis of its own local perception of the world.

One area of interest within computer animation is the description and control of all types of motion. Computer animators seek both to invent wholly new types of abstract motion and to duplicate (or make variations on) the motions found in the real world. At first glance, producing an animated, computer graphic portrayal of a flock of birds presents significant difficulties. Scripting the path of a large number of individual objects using traditional computer animation techniques would be tedious. Given the complex paths that birds follow, it is doubtful this specification could be made without error. Even if a reasonable number of suitable paths could be described, it is unlikely that the constraints of flock motion could be maintained (for example, preventing collisions between all birds at each frame). Finally, a flock scripted in this manner would be hard to edit (for example, to alter the course of all birds for a portion of the animation). It is not impossible to script flock motion, but a better approach is needed for efficient, robust, and believable animation of flocks and related group motions.

This paper describes one such approach. This approach assumes a flock is simply the result of the interaction between the behaviors of individual birds. To simulate a flock we simulate the behavior of an individual bird (or at least that portion of the bird's behavior that allows it to participate in a flock). To support this behavioral "control structure," we must also simulate portions of the bird's perceptual mechanisms and aspects of the physics of aerodynamic flight. If this simulated bird model has the correct flock-member behavior, all that should be required to create a simulated flock is to create some instances of the simulated bird model and allow them to interact.**

Some experiments with this sort of simulated flock are described in more detail in the remainder of this paper. The success

*In this paper *flock* refers generically to a group of objects that exhibit this general class of *polarized, noncolliding, aggregate motion*. The term *polarization* is from zoology, meaning alignment of animal groups. English is rich with terms for groups of animals; for a charming and literate discussion of such words see *An Exaltation of Larks*. [16]

**This paper refers to these simulated bird-like, "bird-oid" objects generically as "boids" even when they represent other sorts of creatures such as schooling fish.

symbolics



symbolics

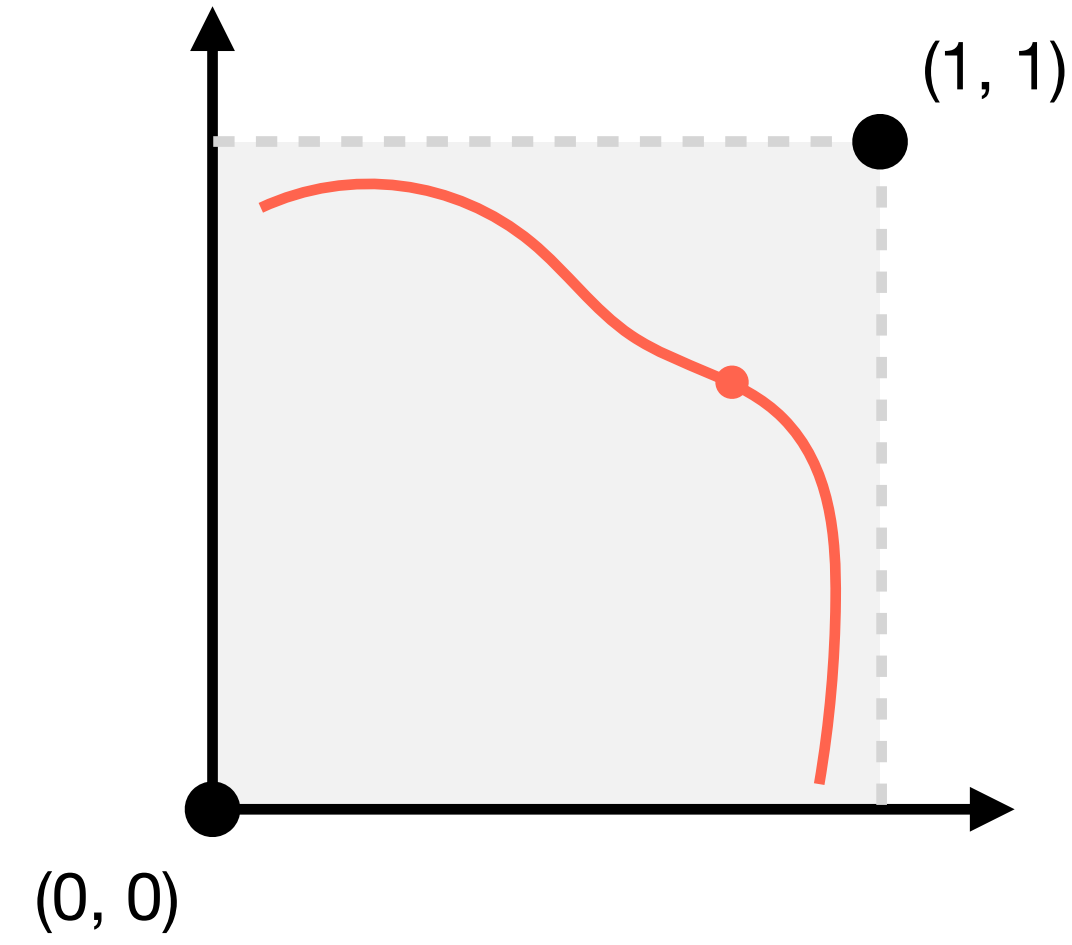
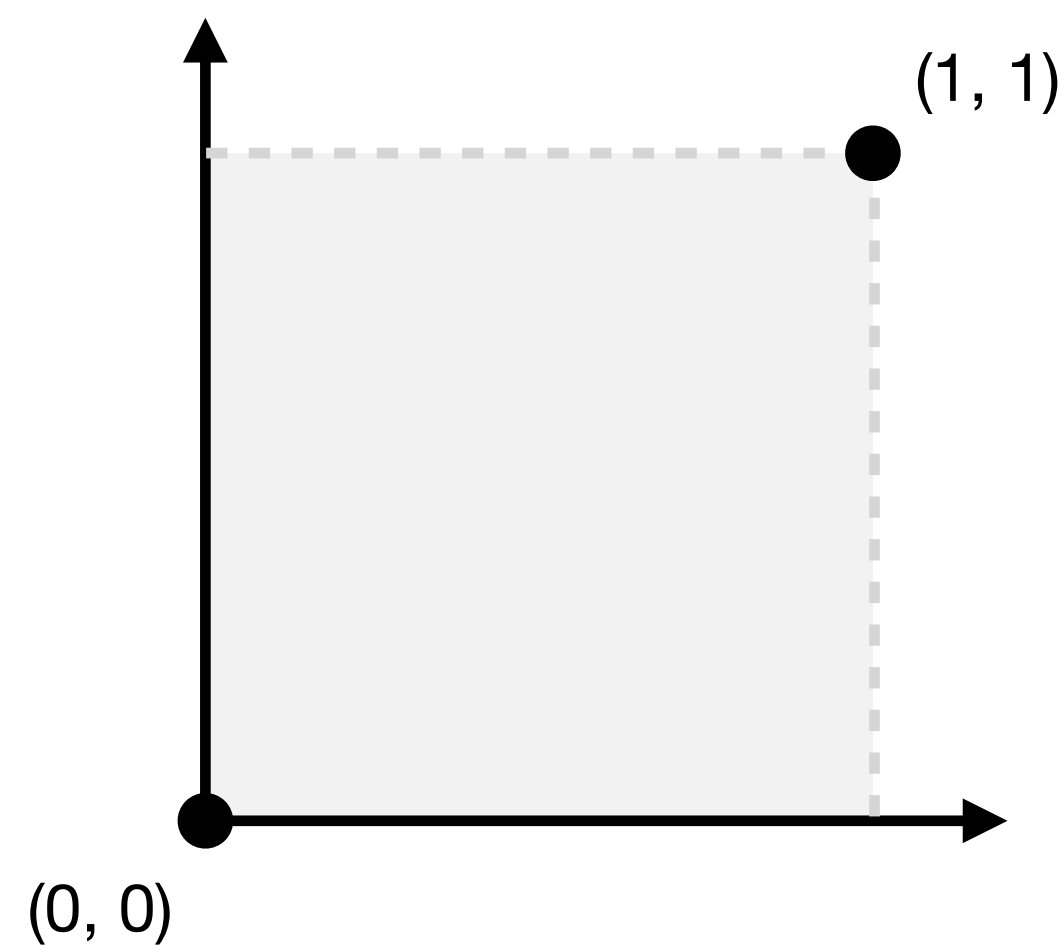


Inverse design of flocks

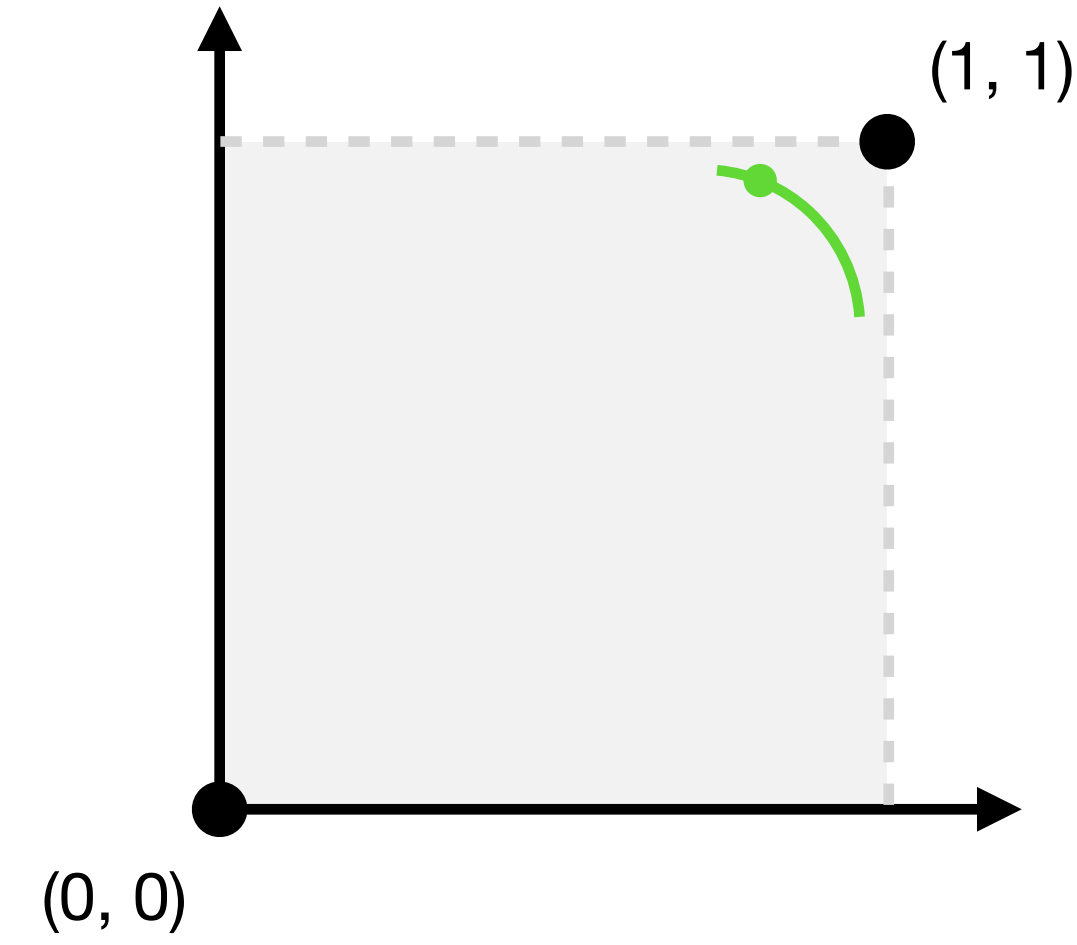
2023-2026

Multi-objective optimization

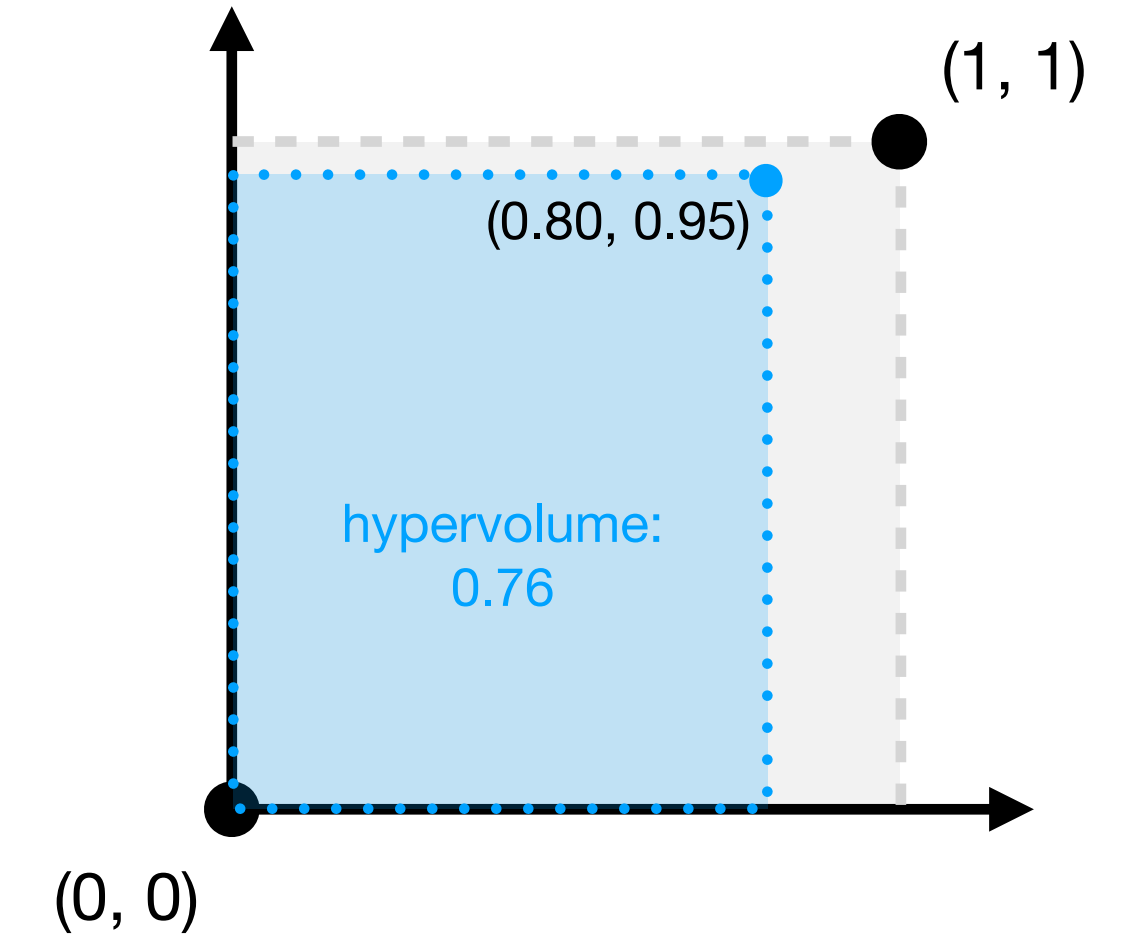
Multi-objective optimization — normalized fitness space.



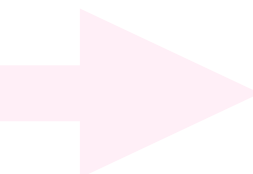
Pareto front for two mutually conflicting objectives.



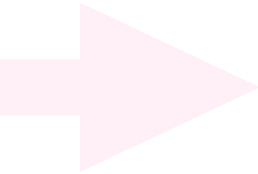
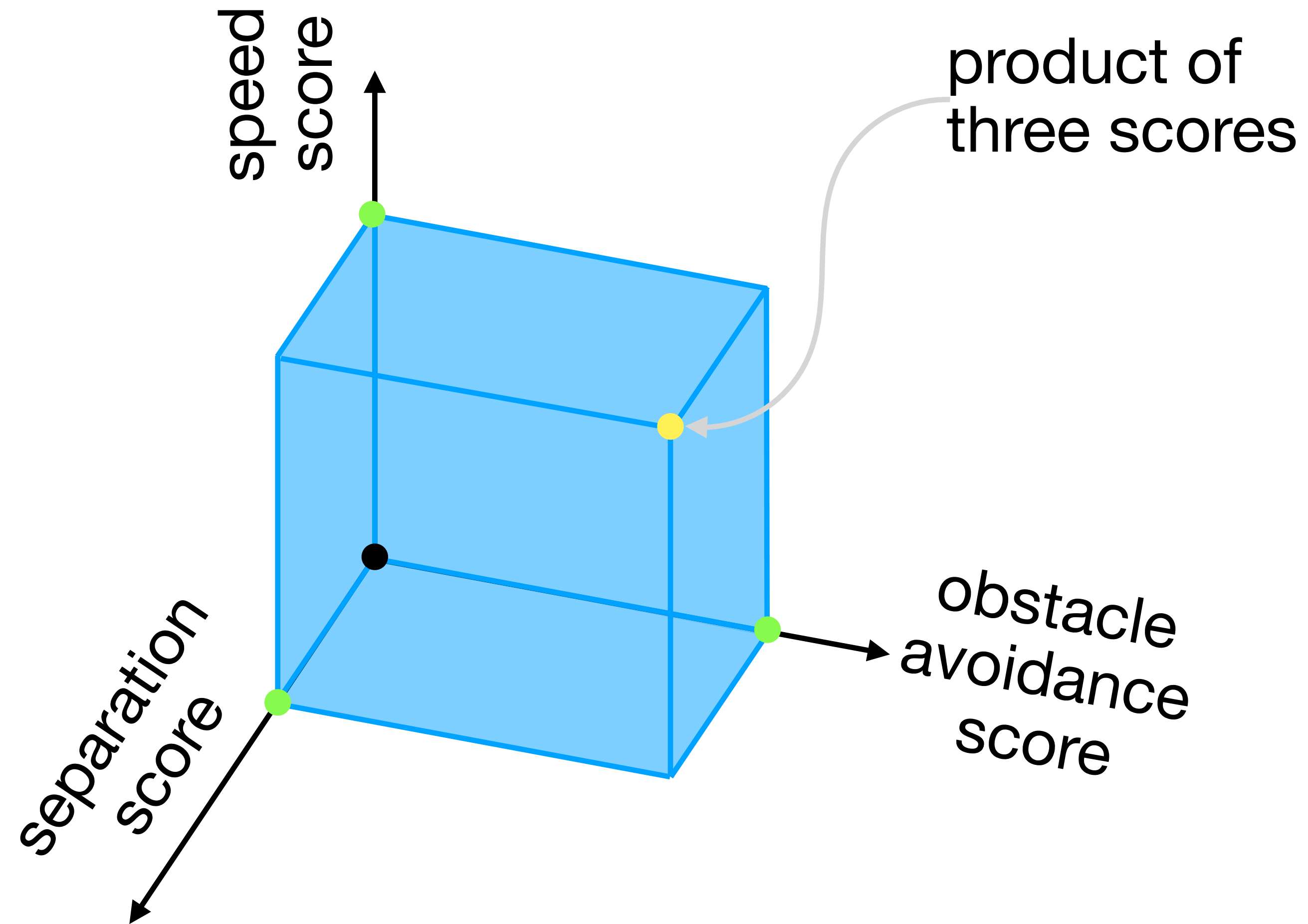
Pareto front for two “mostly compatible” objectives.



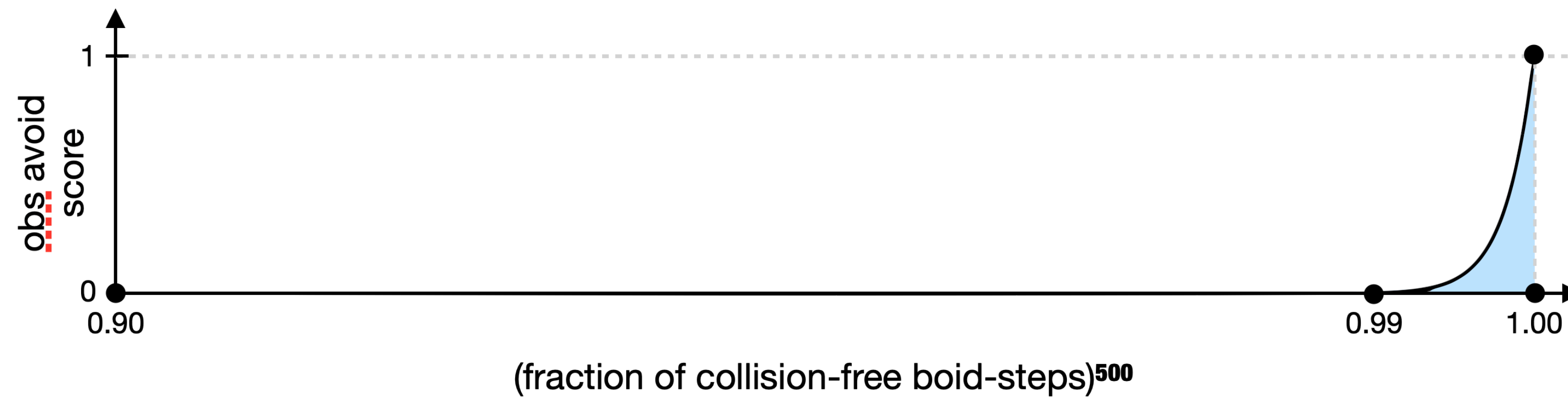
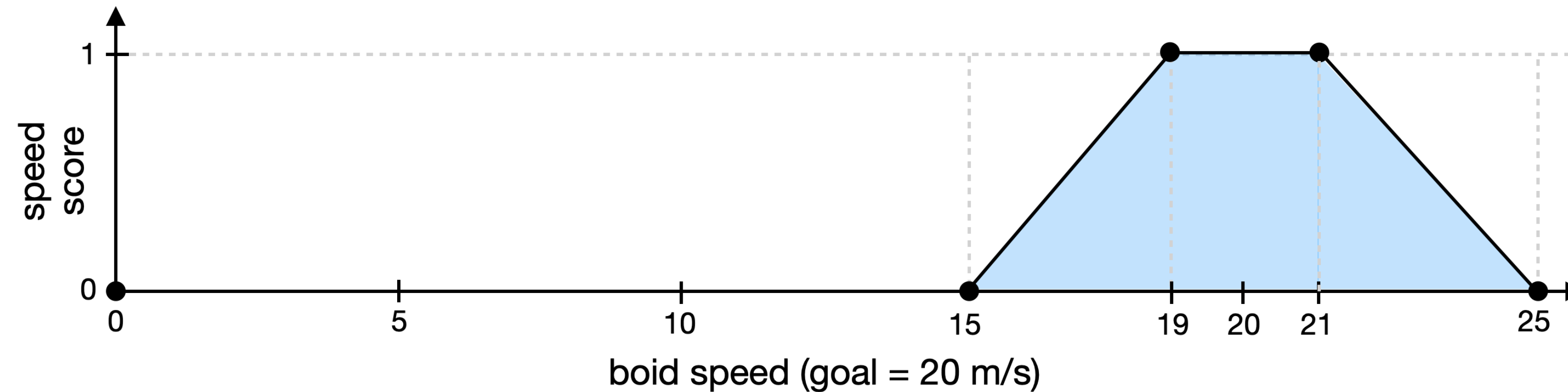
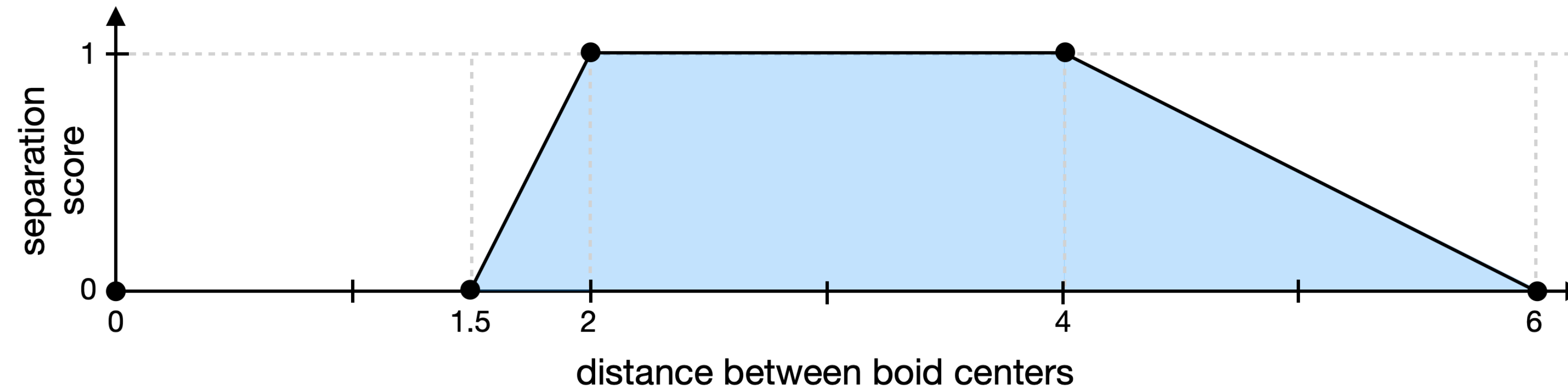
Scalarization (area) for two objectives.



Hypervolume of three objectives



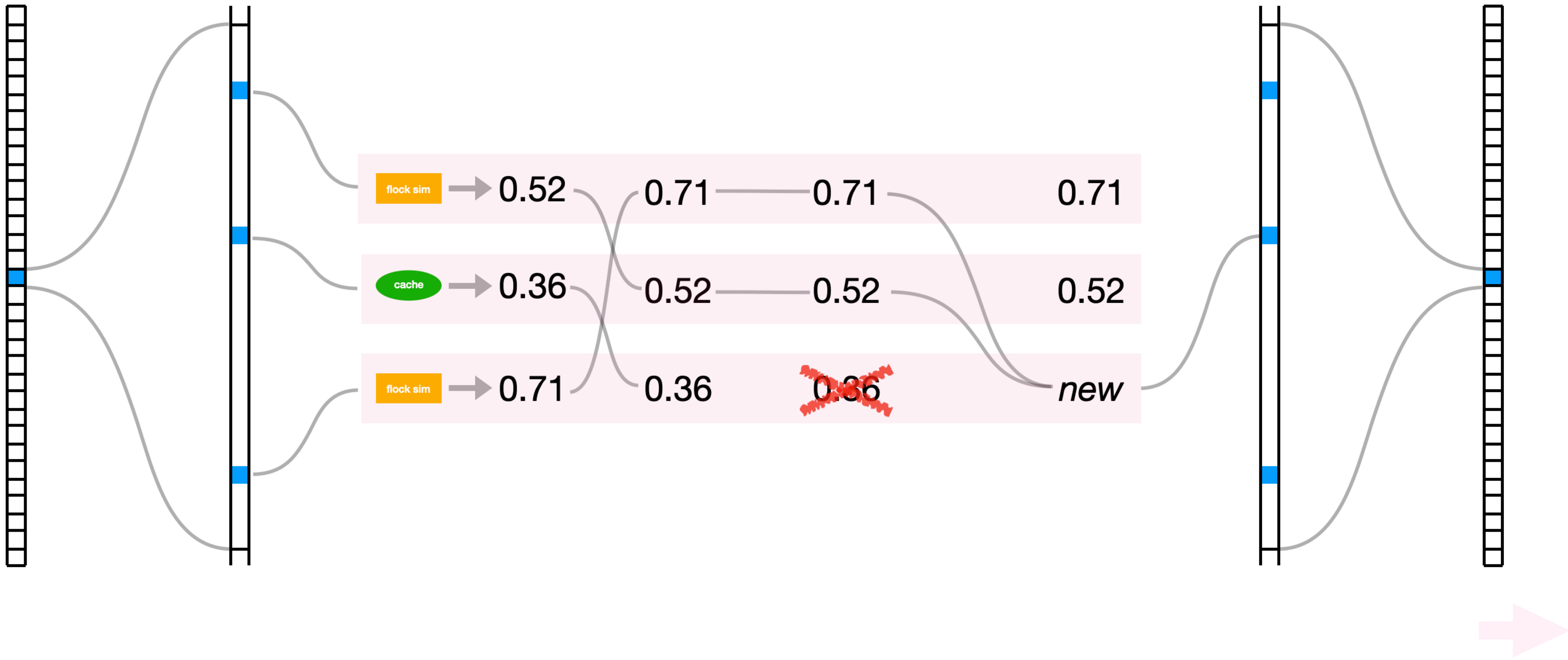
Score weighting functions: separation, speed, avoid



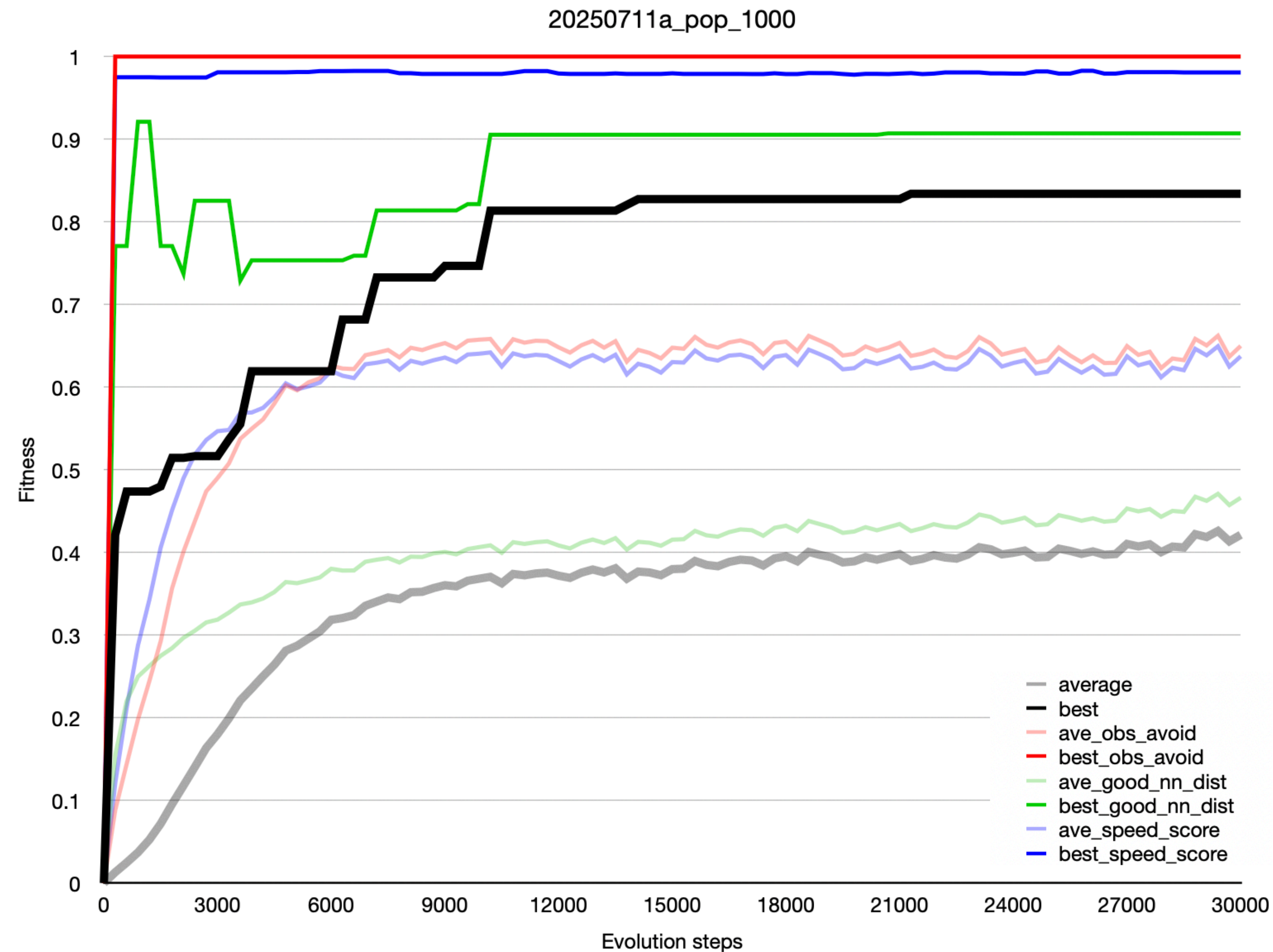
Evolutionary optimization

Single step of steady-state genetic algorithm (SSGA)

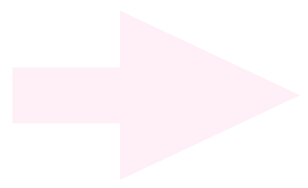
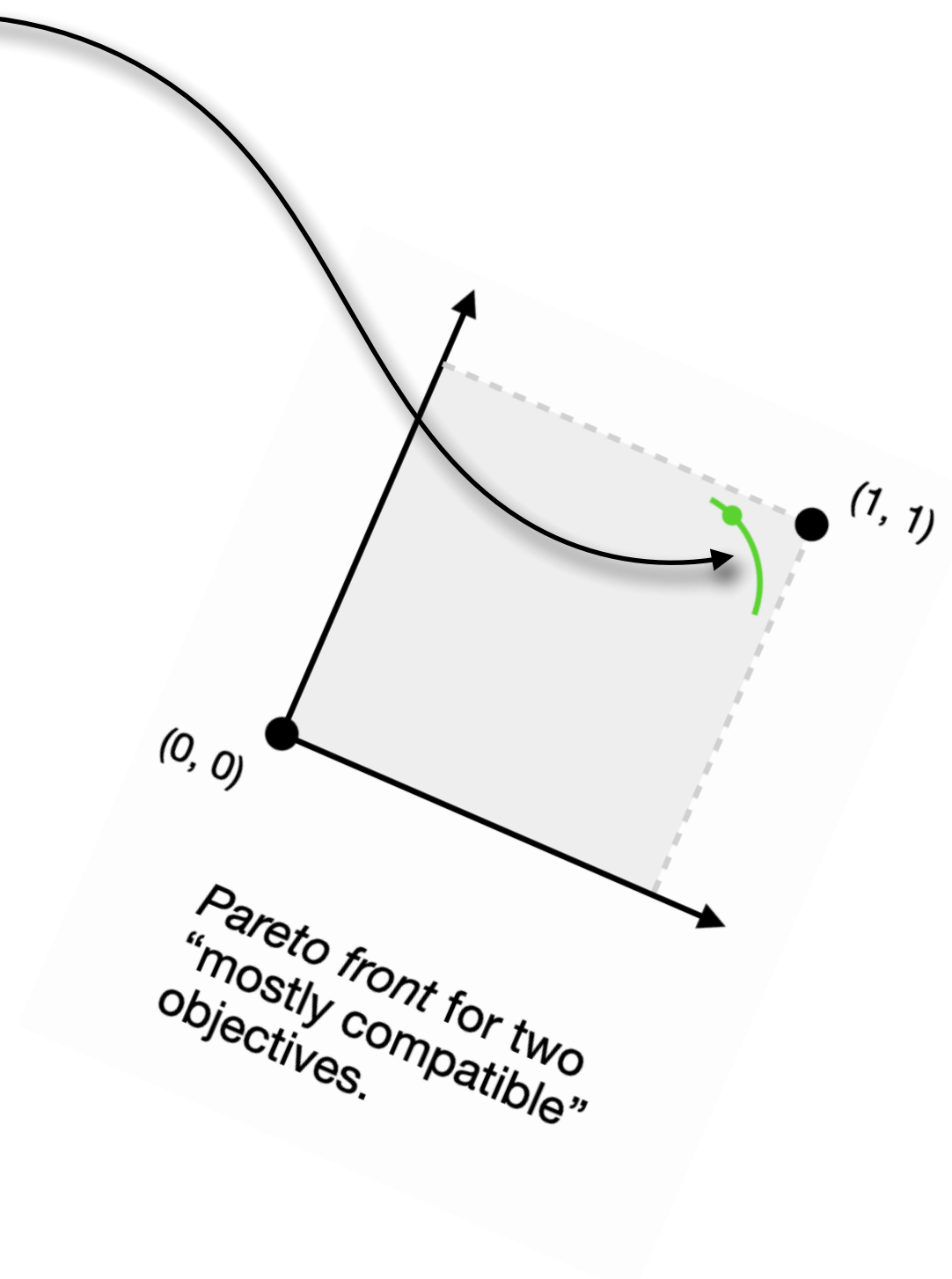
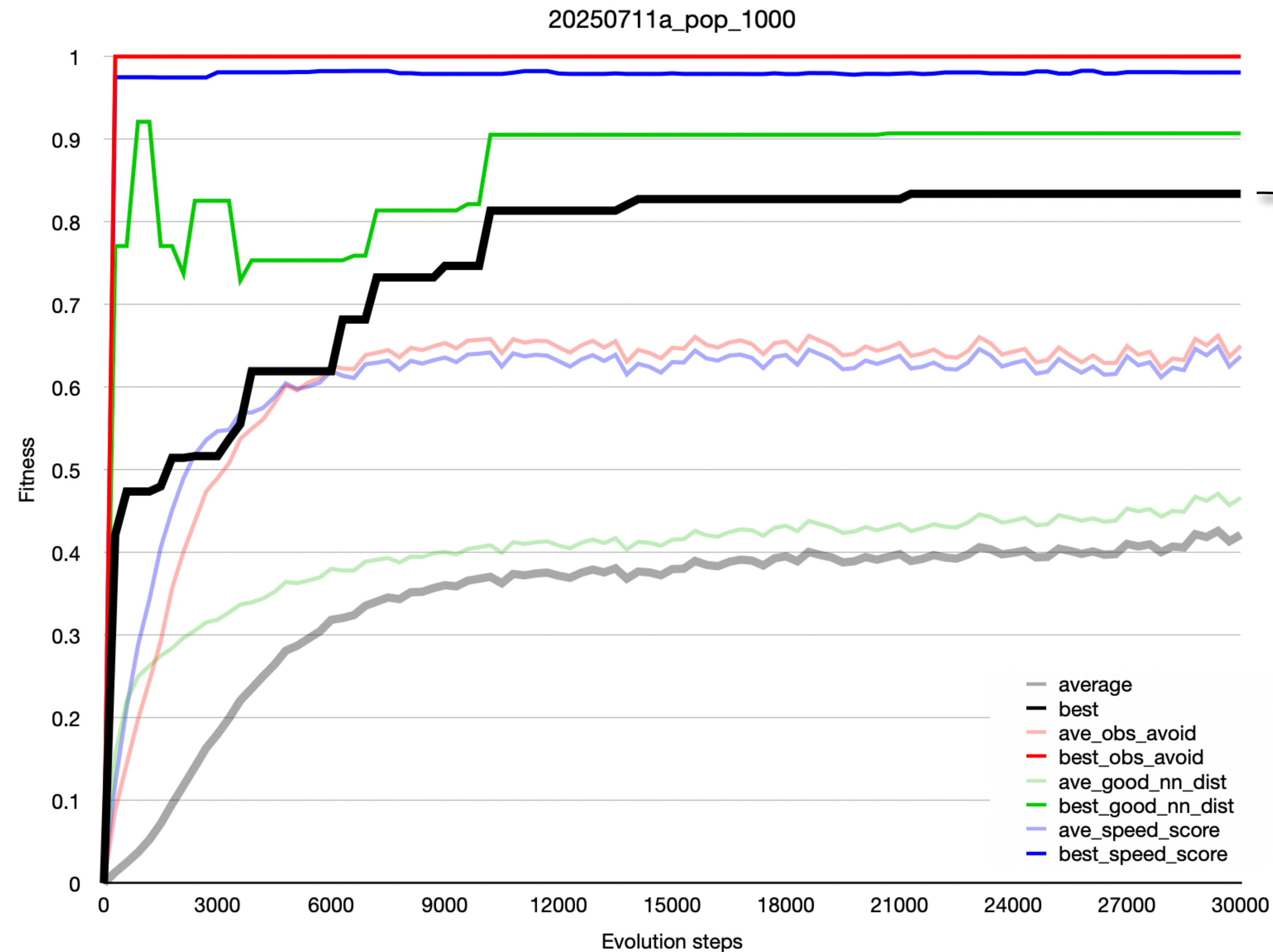
3 way tournament, negative selection, preserves diversity



EvoFlock fitness (and multiple objectives)



EvoFlock fitness (and multiple objectives)



Video 1: basic flocking

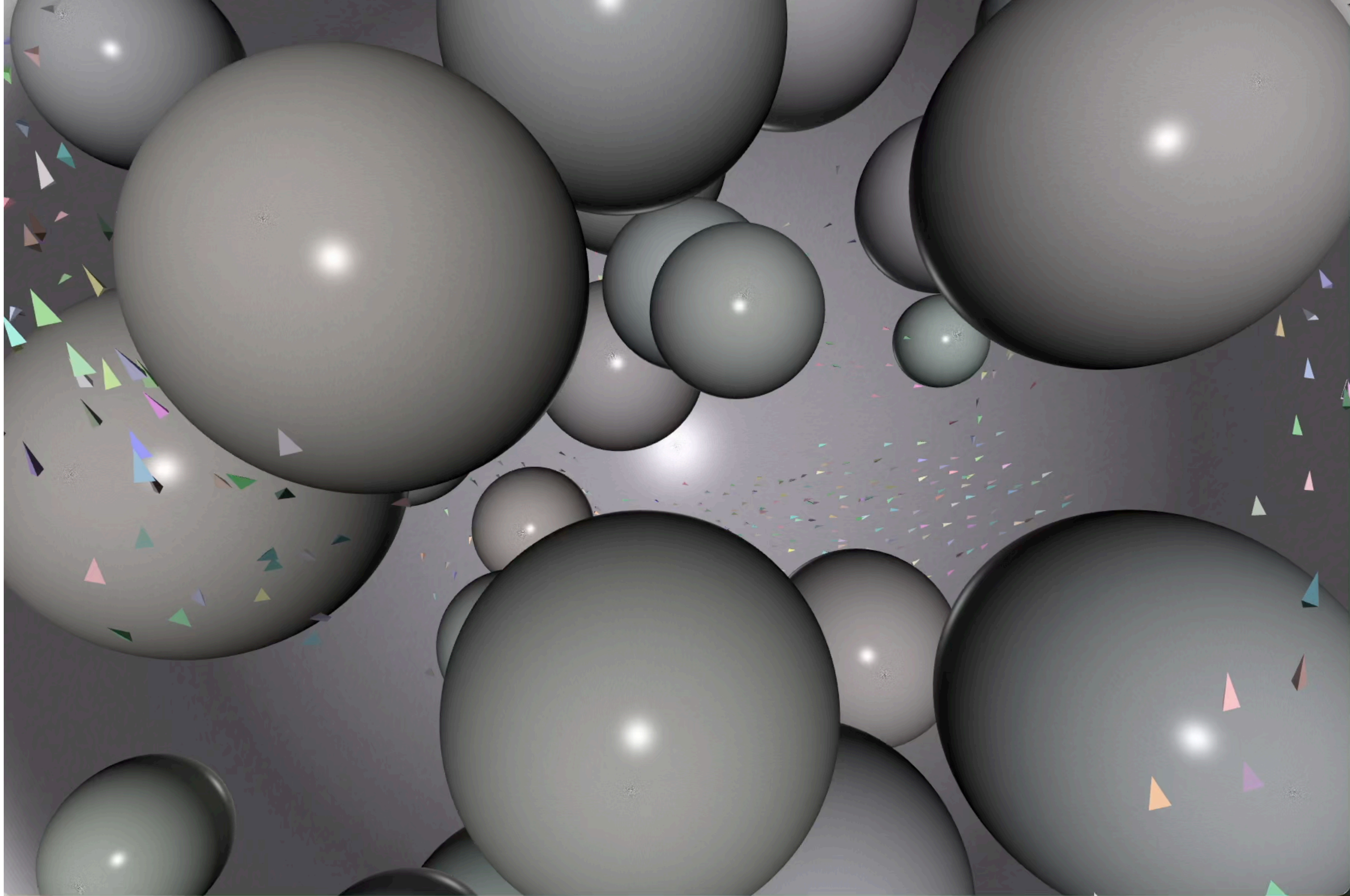
- **Three objectives:**
 - speed within range
 - separation within range
 - avoid (almost all) obstacles



Adding an objective: curvature

Video 2: more curvy

- **Four objectives:**
 - speed within range
 - separation within range
 - avoid (almost all) obstacles
 - **increase curvature**



Ongoing experiments

Unsuccessful tests using genetic programming

Simulating murmurations

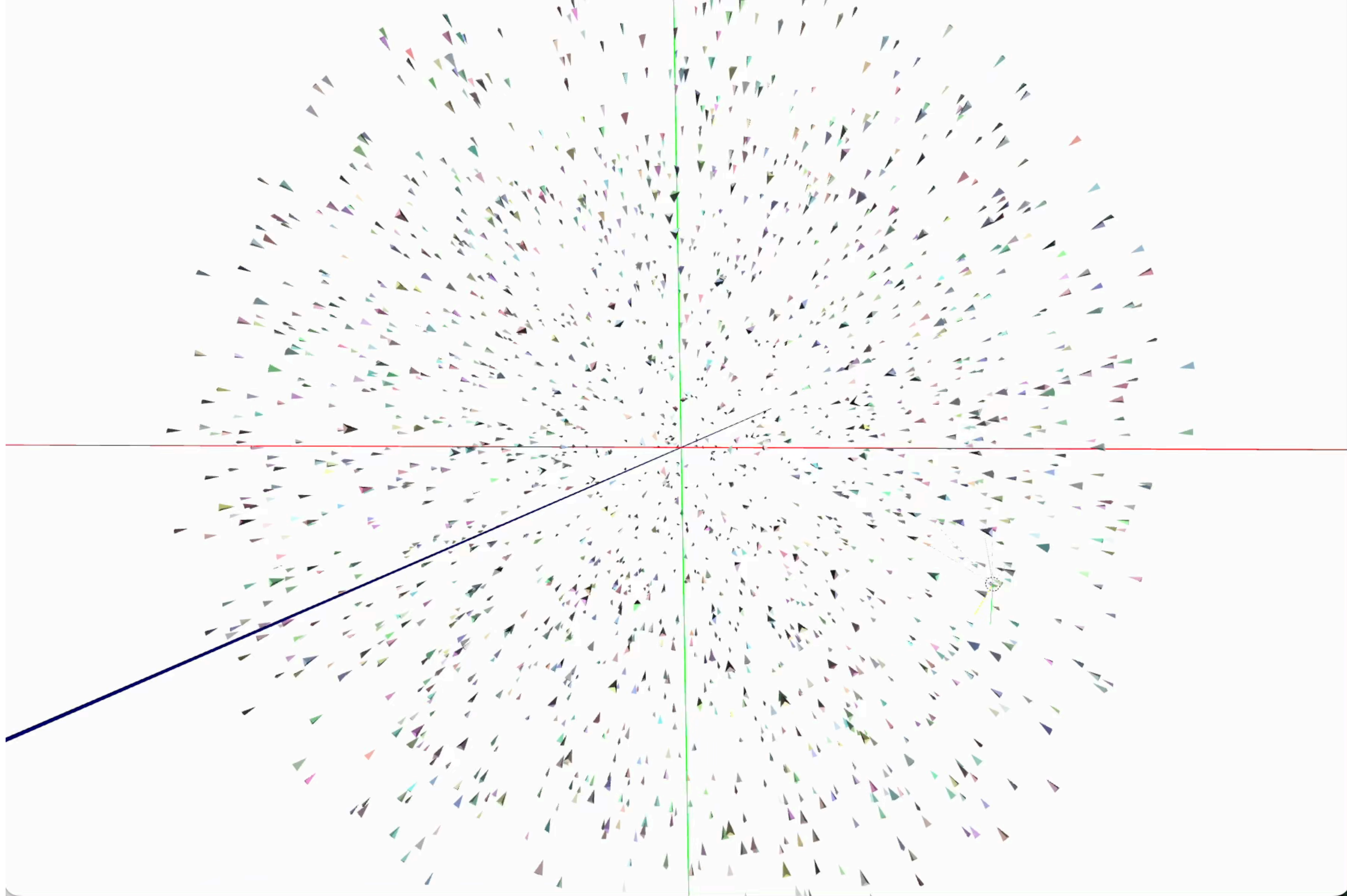
Early work: model of *murmurations*

Without goals or obstacles, sub-flocks form
and head off on non-life-like straight line paths

New steering behaviors

New objective functions

Still very “work in progress”



murmuration test on web



[bird shaped murmuration on YouTube](#)

Thank you!

contact:

cwr@red3d.com

<https://www.red3d.com/cwr/>

these slides



preprint paper

