Spatial Vectorization

This lesson discusses spatial vectorization, along with an interesting case study on Boids.

What is Spatial Vectorization?

> *Spatial vectorization* refers to a situation where elements share the same computation but are in interaction with only a subgroup of other elements.

This was already the case for the game of life example, but in some situations there is an added difficulty because the subgroup is dynamic and needs to be updated at each iteration. This is the case, for example, in particle systems where particles interact mostly with local neighbors. This is also the case for "*boids*" that simulate flocking behaviors.

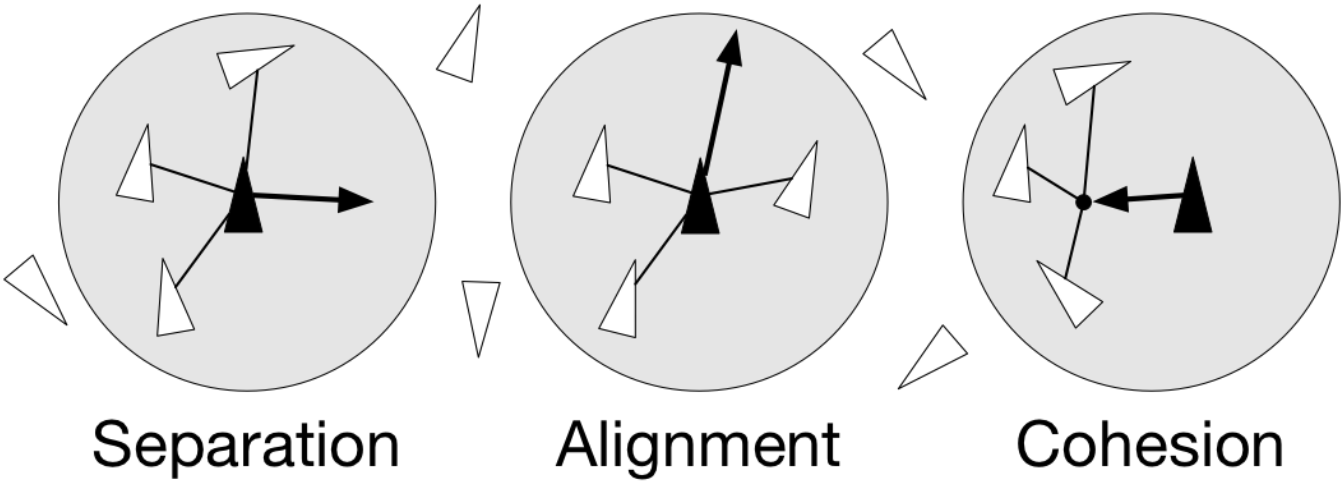
Flocking birds are an example of self-organization in biology. Image by Christoffer A Rasmussen (https://commons.wikimedia.org/wiki/File:Fugle,_ørnsø_073.jpg), 2012.

Problem Description

Boids is an artificial life program, developed by Craig Reynolds in 1986, which simulates the flocking behaviour of birds. The name "boid" corresponds to a shortened version of "bird-oid object", which refers to a bird-like object. — *An Excerpt from the Wikipedia entry Boids*

As with most artificial life simulations, Boids is an example of emergent behavior; that is, the complexity of Boids arises from the interaction of individual agents (the boids, in this case) adhering to a set of simple rules. The rules applied in the simplest Boids world are as follows:

- **Separation:** steer to avoid crowding local flock-mates
- **Alignment:** steer towards the average heading of local flock-mates
- **Cohesion:** steer to move toward the average position (center of mass) of local flock-mates



Boids are governed by a set of three local rules (separation, cohesion and alignment) that serve as computing velocity and acceleration.

Now that you have learned about Boids, let's try to implement its class using the conventional Pythonic approach!