Illustration of Centralized Command and Control for Flocking Behavior

Sami Oweis, Oakland University, Rochester, MI, USA
Subramaniam Ganesan, Oakland University, Rochester, MI, USA
Ka C Cheok, Oakland University, Rochester, MI, USA

ABSTRACT

Flocking is a term that describes the behavior of a group of birds (a “flock”) in flight, or the swarming behavior of insects. This paper presents detailed information about how to use the flocking techniques to control a group of embedded controlled systems - "Boids" - such as ground systems (robotic vehicles/swarm robots). Each one of these systems collectively moves inside/outside of a building to reach a target. The flocking behavior is implemented on a server-based control, which processes each of the boids’ properties e.g. position, speed & target. Subsequently, the server will assign the appropriate move to a specific boid. The calculated information will be used locally to control and direct the movements/flocking for each boid in the group. A simulation technique and detailed flow chart is presented. In addition to Reynolds three original rules for flocking, two other rules - targeting obstacle avoidance - are presented. Our result shows that the obstacles’ avoiding rule was utilized to ensure that the flock didn’t collide with obstacles in each of the boids’ paths.

Keywords: 2D/3D flocking, Boids Properties, Constraints, flocking, Quad Rotors And Path Planning, Robotic Vehicles, Rules Of flocking, flight, Swarm Robots

INTRODUCTION

Flocking is a behavior that shows a set of independent, self-propelled entities moving together collectively (Reynolds, C., 1987; Oweis, S., Ganesan, S. & Cheok, K., 2014; Wikipedia, n.d.); these entities are called boids (Red3d.com, n.d.). To participate in a flock, each entity must be able to coordinate its own movement while considering the effect on its neighboring entities. An entity cannot move too fast or too slow relative to the flock. It also cannot be too close to or too far from any proximal entities. This requires each entity to continuously adjust its motion so not to disrupt any other neighboring entity. Flocking behavior can be found both in natural and artificial environments. Common natural flocks include bird formations or schools of fish. Artificial flocking behavior can often be found in robotics and aerospace. Examples of artificial applications are controlling the behavior of Unmanned Aerial Vehicles (UAVs), screensavers and movie animation.

DOI: 10.4018/ijhcr.2014040101
In normal flocking behavior, there is no centralized control of the flock as a group. Each bird behaves autonomously. In other words, each bird has to decide for itself which flock to consider as its environment. Usually, environment is defined as a circle (2D) or an orb (3D) with a particular radius, representing the wing reach or domain of an individual bird (Reynolds, C., 1987; Oweis, S., Ganesan, S. & Cheok, K., 2014; Wikipedia, n.d.; Red3d.com, n.d.).

A basic utilization of a flocking algorithm has a time complexity of \( O(n^2) \) (Wikipedia, n.d.) - each bird searches through all other birds to find those who fall into this environment, where \( n \) is the number of the birds in the flock. This paper presents a new way of applying flocking behavior by usage of a central command and control flocking algorithm. The algorithm can run on a server to have overall control of each boid in the flock, and a better path planning for the boid.

This paper is structured as follows: Section II describes the properties of the boids in the flock; Section III describes the original three Reynolds rules, while Section IV includes two addition rules of flocking. They were added to support both obstacle avoidance and targeting. Section V shows implementation of the flocking simulation using MATLAB® software (Mathworks, n.d.). Finally, Section VI provides a conclusion and insight into future work.

**Boids Properties**

There are two sets of properties for the boids: global and environmental parameters. Examples of these are the radius of the environment; the coefficient of friction (in both static friction and kinetic friction); and acceleration. There are other local boid parameters, such as mass, separation, alignment, cohesion, targeting and avoiding strengths, velocity and force limits, in addition to an assigned number ID to uniquely identify each boid in the space.

**Reynolds Rules of Flocking**


**Definitions**

For illustration, suppose a flock consists of five boids having randomly assigned positions P1, P2, P3, P4, and P5, as shown in Table 1 below. A neighborhood flocking radius = 30 meters.

- \( P_i(k) = \begin{pmatrix} px_i(k) & py_i(k) & pz_i(k) \end{pmatrix} \) Position of the \( i^{th} \) boid.
- \( P_n(k) = \begin{pmatrix} px_n(k) & py_n(k) & pz_n(k) \end{pmatrix} \) Position of a neighbor in the set \( N \).

\( N \rightarrow \) Number of boids counted within neighborhood radius of the \( i^{th} \) boid.

\( P_i(k) \neq P_n(k) \) For all \( n \in N \), where \( n \neq i \), \( k \rightarrow \) Time instance.

**Flocking Rule #1: Separation**

This rule ensures that each boid does not get too close to any boid near it. This rule is necessary to prevent crowding. It is being applied by creating a velocity vector, by using the following equation:
Two Faces of Mobile Shopping: Self-Efficacy and Impulsivity
www.igi-global.com/chapter/two-faces-of-mobile-shopping/183309?camid=4v1a

Threat and Risk-Driven Security Requirements Engineering
www.igi-global.com/article/threat-risk-driven-security-requirements/51660?camid=4v1a