

Advanced Movements

Flocking

Manuel Ladebeck

16.05.2005



Inhalt

- 1. Was ist „Flocking“?
- 2. Classic (Basic) Flocking
- 3. Obstacle Avoidance
- 4. Follow the leader
- 5. Demo
- 6. Anmerkungen/Ausblick
- 7. Quellen

Was ist „Flocking“?

- Es sieht unnatürlich aus, wenn jeder in einer Gruppe von Einheiten nur seinen eigenen Weg geht
- Gruppenmitglieder sollten Einfluss auf die Bewegung haben
- Flocking ist ein Verfahren, um Gruppendynamik zu simulieren

Wozu Flocking?

Vielerlei Anwendungsgebiete:

- Schwarm von Vögeln
- Fische
- Militärische Einheiten
- Menschenmassen
- usw.

Geschichte des Flockings

Die Idee und erste Umsetzung stammt von Craig Reynolds, erstmals veröffentlicht in seinem 1987 erschienenem Paper „Flocks, Herds, and Schools: A Distributed Behavioral Model“ .

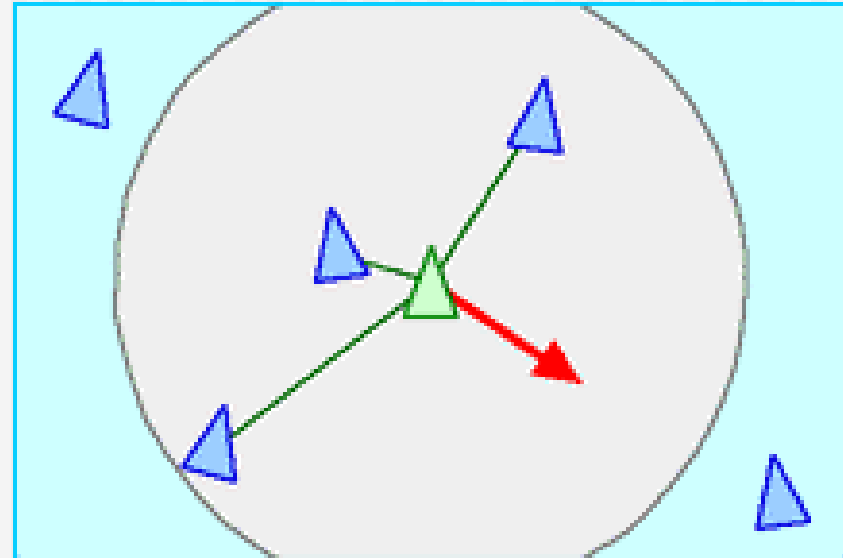
Dieses Paper präsentiert einen Algorithmus, der zu sehr natürlichem Bewegungsverhalten führt.

Classic Flocking

- Idee von Reynolds:
Eigene Bewegung wird von Nachbarn beeinflusst
- Prägte den Begriff „Boids“
- Sehr natürliche Bewegung durch folgende 3 Regeln:
 - ✓ Separation (Abstand)
 - ✓ Alignment (Ausrichtung)
 - ✓ Cohesion (Zusammenbleiben)

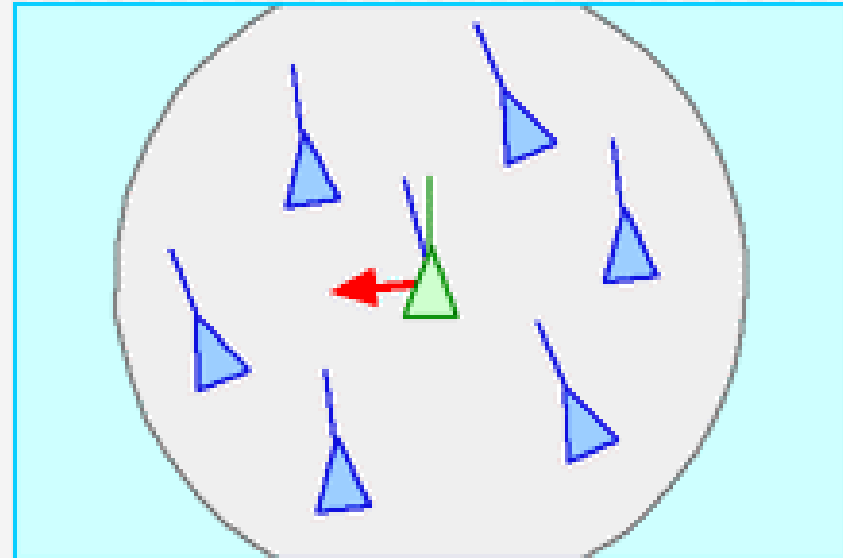
Separation

- Steuere so, dass du nicht mit deinen Nachbarn kollidierst



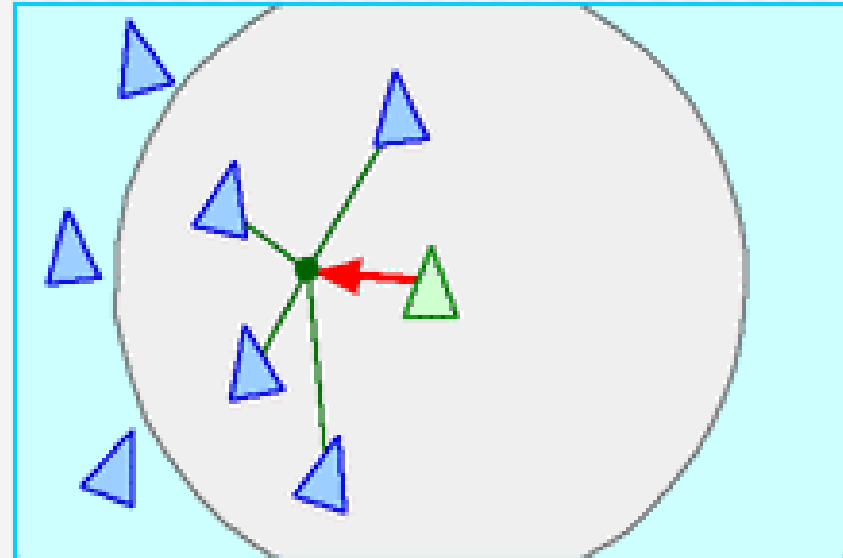
Alignment

- Steuere so, dass deine Ausrichtung dem Durchschnitt deiner Nachbarn entspricht



Cohesion

- Steuere in Richtung der durchschnittlichen Position deiner Nachbarn

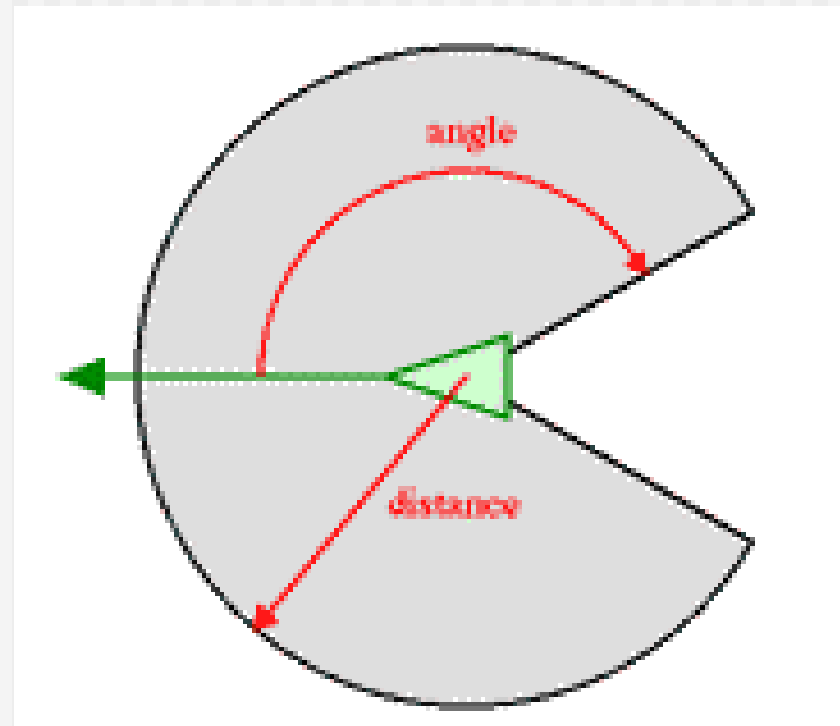


Voraussetzungen

- Jede Einheit muss steuern können, z.B. über einen Geschwindigkeits- und Richtungsvektor (siehe Vortrag „Chasing and Evading“)
- Die drei aus den Regeln resultierenden Kräfte müssen zueinander gewichtet werden
- Jede Einheit muss ihre Nachbarn kennen

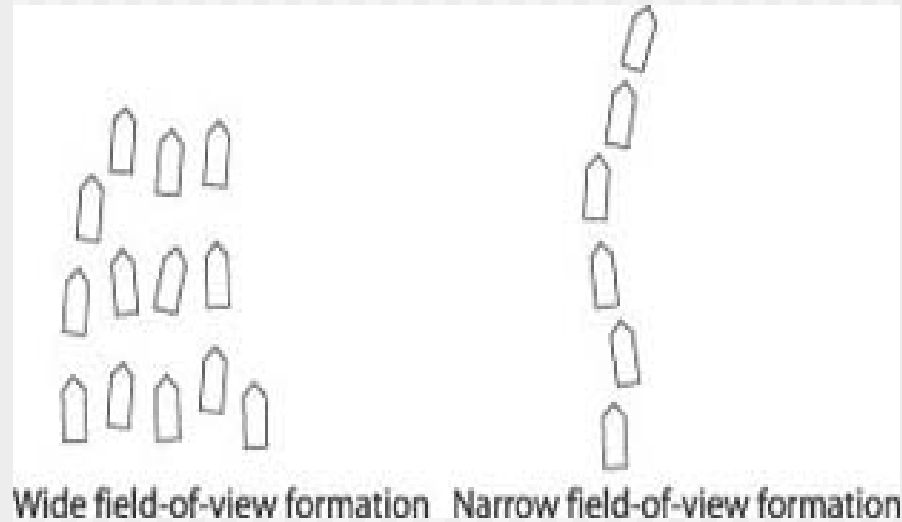
Erkennen der Nachbarn

- Jede Einheit hat einen Wahrnehmungsbereich, bestimmt durch einen Radius und einen Winkel
- Anpassbar, je nach gewünschtem Verhalten
- Jedoch: zu großer Winkel führt zu unnatürlichem Verhalten!



Radius und Winkel

- Großer Radius => Gruppe tendiert dazu enger zusammenzubleiben
- Kleiner Radius => Es bilden sich mehrere kleine Splittergruppen
- Winkel: siehe Grafik



270°

45°

Feintuning

- Erfordert Feintuning durch Trail and Error
 - Keine einzelne Kraft darf dauerhaft dominieren
- ⇒ Intelligente Gewichtung der Kräfte vonnöten

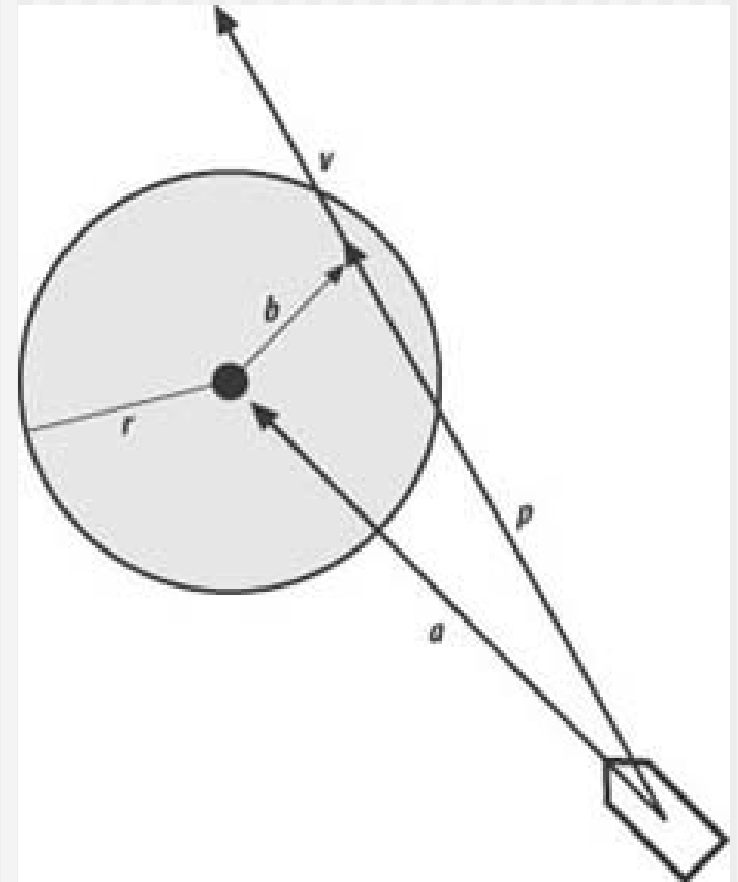
Zwischenbilanz

Das leistet der Basic Flocking Algorithmus bisher:

- Natürliche Bewegung von beliebig großen Gruppen
- Anpassbar durch Sichtbereich
- Doch in den meisten Spielen gibt es Hindernisse, wie gehen wir damit um?

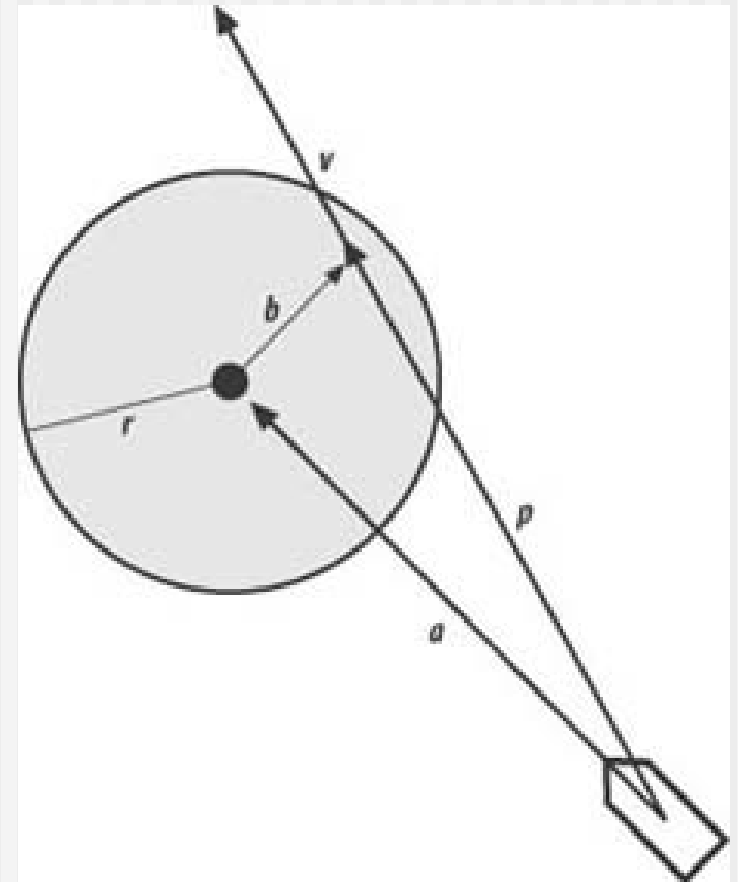
Obstacle Avoidance

- Simpel einzubauen, da: einfach eine weitere Steuerungskraft
- Grundidee: Jede Einheit erhält einen „Fühler“, um Hindernisse vor der Kollision zu entdecken
- Der Fühler ist ein Vektor



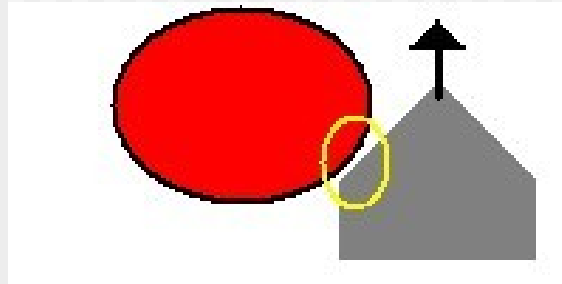
Obstacle Avoidance (2)

- Stößt der Vektor auf ein Hindernis, berechnet man eine Steuerungskraft weg vom Hindernis
- Funktioniert gut, aber vermeidet nicht 100% aller Kollisionen



Obstacle Avoidance (3)

- Breitere Einheiten brauchen möglicherweise mehrere Fühler, sonst:



- Menge von anderen Ansätze zur Kollisionserkennung sind möglich

Follow the leader

- Idee: Es gibt einen Anführer der Gruppe, der nicht der Gruppendynamik unterliegt
- Alle anderen Gruppenmitglieder unterliegen einer weiteren Steuerungskraft, die sie dem Anführer folgen lässt

Follow the leader (2)

Viele praktische Anwendungsgebiete in Videospielen:

- Flugsimulationen => Wingmen
- Taktikshooter => Kompanie, die einem Anführer durch den Dschungel folgt

Follow the leader (3)

- Der Anführer selbst unterliegt wiederum einer beliebigen Anzahl von Steuerkräften

Beispielsweise:

- Ziel verfolgen (chase/intercept)
- Fliehen (evade)
- Punkt (x,y) erreichen

Flocking – Eine Implementierung

- Implementierung der vorgestellten Konzepte
- Umfasst den Basic Flocking Algorithmus, sowie Obstacle Avoidance und Follow-the-leader

[Zeige Flocking Demo](#)

Kommerzielle Spiele mit Flocking



- „Half-Life“ nutzt Flocking zur Bewegung der Marines, die den Spieler auf verschiedene Arten koordiniert angreifen

Kommerzielle Spiele mit Flocking (2)



- „Enemy Nations“ verwendet modifizierte Flocking Algorithmen, um die Bewegung und Formationen von Einheitengruppen zu koordinieren

Kommerzielle Spiele mit Flocking (3)



- „Unreal“ nutzt Flocking für verschiedene Monster, sowie Fische und Vögel

Zusammenfassung

- Flocking Algorithmen liefern auf relativ einfache Art eine natürliche Gruppenbewegung und tragen damit zur Glaubhaftigkeit der Spielewelt bei
- Leicht erweiterbar und anpassbar
- In vielen verschiedenen Spielen einsetzbar

Anmerkungen

- Straight-forward Implementierung hat Laufzeit $O(n^2)$
- Kann optimiert werden, z.B. durch Verwendung einer Datenstruktur, bei der die Einheiten nach ihrer Position sortiert sind
- Mit solchen Optimierungen und moderner Hardware können riesige Gruppen simuliert werden
- Es gibt kein Allround Algorithmus, sollte stets an das jeweilige Problem angepasst werden

Ausblick

- Höherer Stellenwert der KI in Zukunft => Flocking Algorithmen können noch mehr Verwendung finden
- Filmindustrie => Computer animierte Filme boomen

Quellen

- Craig Reynolds: Flocks, Herds, and Schools: A Distributed Behavioral Model

<http://www.cs.toronto.edu/~dt/siggraph97-course/cwr87/>

- Boids: <http://www.red3d.com/cwr/boids/>
- AI for Game Developers by David M. Bourg, Glenn Seeman
- <http://drhuxtable0.tripod.com/> HL screenshot
- <http://www.enemynations.com/> EN screenshot
- <http://www.oreilly.com/catalog/ai/> AIDemo Source Code