

Flocking

Autor: Diego G. Ruiz

Versión: 1.0 (draft)

Tabla de Contenidos

Introducción.....	3
Su aplicación en juegos.....	3
Reglas básicas.....	4
Reglas extendidas.....	4
Agentes inteligentes.....	5
Implementación.....	6
Obtención de las entidades locales.....	6
Toma de decisión.....	7
Separación.....	7
Alineamiento.....	7
Cohesión.....	8
Reglas adicionales.....	8
Limitador de aceleración.....	8
Contención dentro de un volumen.....	8
Persecución de metas.....	9
Conclusión.....	9
Bibliografía.....	10

Flocking

Introducción

Flocking es el nombre que se le da al modelo computacional utilizado para el movimiento coordinado de grupos (flocks) de entidades (boids). Este movimiento coordinado puede apreciarse, en la naturaleza, en el vuelo de los pájaros, peces, insectos, etc.

Fue introducido por primera vez por Craig Reynolds por medio de su paper "Flocks, Herd, and Schools: A Distributed Behavioral Model" en la SIGGRAPH del año 1987.

El modelo posee ciertas reglas que hay que respetar en la implementación, es decir, flocking no es sólo intentar simular el movimiento de entidades coordinadas sino además la aplicación de ciertas reglas que aseguren las condiciones reales de este fenómeno. No puede existir una inteligencia superior que gobierne todas las entidades, cada entidad se mueve de modo independiente según las reglas del modelo y su ponderación respectiva.

De este modo, la simulación que observaremos en pantalla se asemejará mucho al comportamiento de las entidades que podamos observar en la naturaleza. En principio, parece reinar el caos, pero si la observación es detallada se podrá notar que existe un patrón. Las entidades mantienen una cohesión pero al mismo tiempo pareciera existir una distancia mínima entre ellas, no se chocan entre sí ni contra obstáculos (o al menos la posibilidad que esto ocurra es baja) y, finalmente, mantienen una dirección común.

La rama de la inteligencia artificial que se encarga del estudio de este modelo es la vida artificial o A-Life.

De la aplicación de las siguientes reglas sencillas, se obtienen diversos comportamientos complejos (en función de su ponderación, es decir, del peso o la importancia que posee cada una de ellas en la dirección/velocidad/aceleración que finalmente efectuará la entidad). Estos comportamientos se los suele denominar **emergentes**, pues no están planeados a priori y en muchas ocasiones sorprenden al creador del modelo en cuestión.

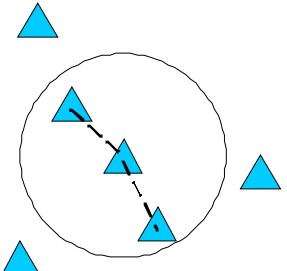
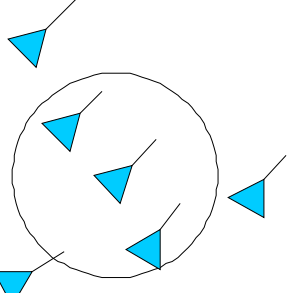
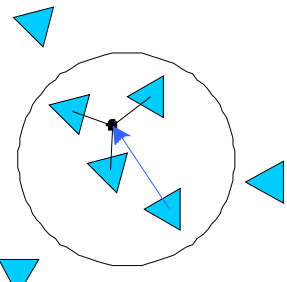
Otro rasgo fundamental de la aplicación de modelos de flocking es la **falta de predictibilidad** del comportamiento del mismo en un lapso de tiempo considerable.

Su aplicación en juegos

Son muchas las aplicaciones posibles de este modelo en juegos de video. Una de ellas es el comportamiento de un conjunto de enemigos (naves espaciales, soldados, insectos, etc.). El uso de flocking, a diferencia del movimiento conjunto monótono, brindará una sensación de realismo muy superior, culminando en un gameplay más profundo y variado.

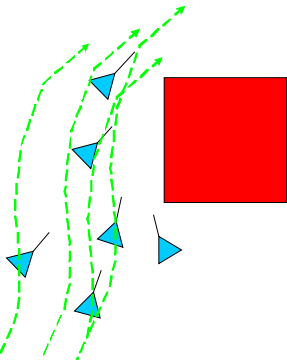
Reglas básicas

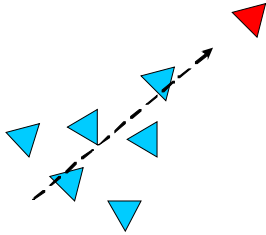
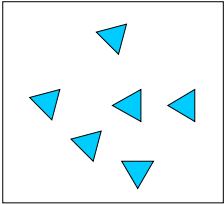
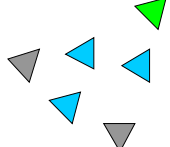
Las consideradas reglas "básicas" son las expuestas inicialmente por Reynolds.

<p>Separación: Las entidades mantienen una separación mínima con sus vecinas.</p>	
<p>Alineamiento: Las entidades mantienen una dirección común. Esta dirección se brinda por medio de un promedio calculado para una entidad por medio del promedio de la dirección de sus vecinas. Luego la dirección de grupo es el promedio total.</p>	
<p>Cohesión: Las entidades se mantienen unidas. Es decir, que existe una separación máxima que no se supera.</p>	

Reglas extendidas

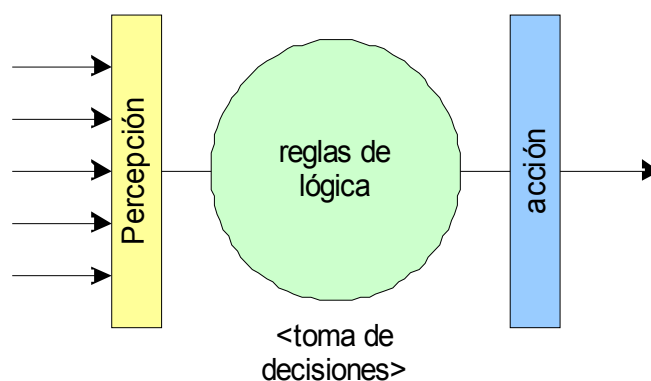
A las reglas básicas de Reynolds se agregan otras que profundizan la simulación:

<p>Esquive: Las entidades esquivan obstáculos.</p>	
---	--

<p>Persecución de metas: Es posible, y muy común, incluir dentro del modelo reglas adicionales relacionadas a predadores que modifiquen el comportamiento de las entidades: ya sea cazando la presa o huyendo del predador.</p> <p>La dirección en principio aleatoria que podría tomar el conjunto se verá modificada por el seguimiento de la presa hasta que la misma sea exterminada, luego el comportamiento del grupo retornará al patrón usual.</p>	
<p>Contención dentro de un volumen: Se especifica un volumen dentro del cual deben permanecer todos los boids.</p>	
<p>Tiempo de vida: Se especifica un tiempo de vida, el cual afecta las propiedades de cada boid (ej: velocidad); llegado un determinado tiempo el boid puede morir.</p>	

Agentes inteligentes

Un agente es una entidad que percibe su entorno y actúa en consecuencia decidiendo que hacer por medio de reglas de lógica propias del agente.



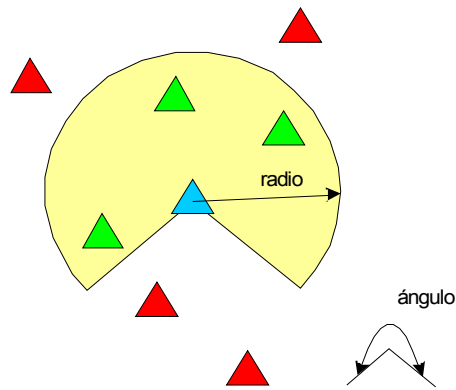
Teniendo en cuenta esta definición cada entidad del grupo o bandada deberá actuar ser un agente inteligente. De este modo, cada entidad percibirá su entorno mediante cierta información acotada disponible para él; acotada porque nuestras entidades simulan seres vivos con organismos de percepción no ideales, con ojos que poseen un cierto campo de visión, con oídos que escuchan hasta una determinada distancia máxima.

Debido a que en nuestro modelo matemático estas "limitaciones" no estarían presentes, acotaremos de forma adrede la información disponible para cada entidad, para que el comportamiento final sea lo más fiel posible al del ser vivo que se intente copiar.

Implementación

Obtención de las entidades locales

Por lo tanto, en la implementación de ciertas reglas deberemos sólo tener en cuenta las entidades vecinas, aquellas que se encuentren dentro del siguiente radio y ángulo:



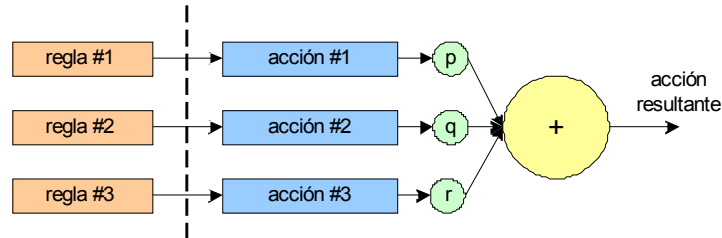
Tanto el radio como el ángulo (usualmente 300 grados) podrían ser parametrizados para luego poder modificarlos y observar cuál es la variación en el comportamiento de la bandada.

Sería grandioso contar con un sistema de análisis de imágenes y que cada entidad obtenga la información visual de las inferencias obtenidas a partir de las mismas. Pero esto sería muy costoso y posiblemente más imperfecto que la visión real del ave. En lugar de ello, sabremos si una entidad está dentro de su "campo de visión" analizando su posición contra la posición de la entidad actual. La distancia entre ambas entidades deberá ser menor al radio correspondiente y el ángulo también deberá estar en el rango especificado.

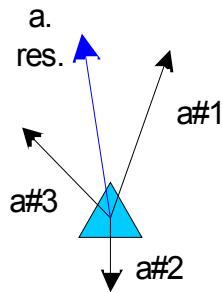
Boids			
boid #1	x	y	z
boid #2	x	y	z
boid #3	x	y	z
boid #4	x	y	z
boid #5	x	y	z
boid #6	x	y	z
boid #7	x	y	z

Toma de decisión

La aplicación de cada una de las reglas nos indica hacia donde movernos. Luego la toma de decisión que tendrá que realizar la entidad será la promediación ponderada de cada una de las acciones sugeridas por las reglas.



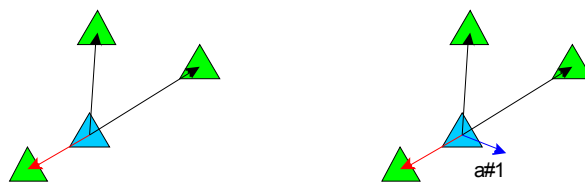
Siendo la acción una aceleración vectorial:



Separación

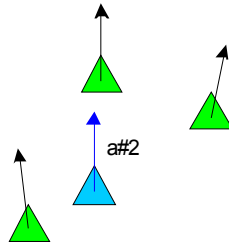
Por lo tanto, implementaremos cada reglas de nuestras entidades de modo independiente y luego decidiremos de que modo se combinará la decisión obtenida por cada una de ellas.

Regla: Verificar distancia contra toda entidad vecina. Si alguna distancia es menor a la mínima estipulada efectuar un movimiento que revierta esta situación.



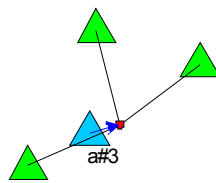
Alineamiento

Regla: Promediar aceleración de todas las entidades vecinas.



Cohesión

Regla: Calcular el punto equidistante a todas las entidades vecinas y dirigirse hacia allí.



Reglas adicionales

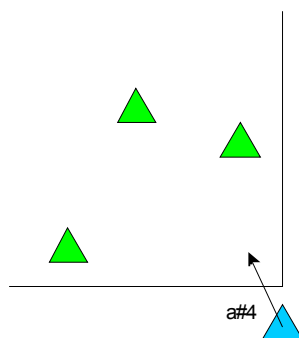
Luego de aplicar las reglas básicas de Reynolds, es posible agregar la cantidad de reglas que el programador considere adecuado. Por lo general, el comportamiento adecuado se obtiene mediante prueba y error, agregando y quitando reglas o modificando los pesos de cada una de las reglas.

Limitador de aceleración

Regla: Si el módulo de la aceleración es mayor a un número especificado limitar dicho valor a un máximo establecido.

Contención dentro de un volumen

Regla: Si la posición de la entidad se encuentra fuera de un determinado volumen, realizar un movimiento de alta prioridad para ingresar nuevamente a él.



Persecución de metas

La dirección a tomar puede ser alterada por una regla de alta prioridad que indique la persecución de una meta. Un caso normal es esta situación es el avistamiento de una presa. Cuando se da esta situación, todas las entidades modifican su dirección para ir detrás de ella, hasta que la misma escapa o es devorada, momento en el cual la regla se desactiva y deja de tener influencia sobre la decisión tomada.

Conclusión

El modelo de comportamiento distribuido introducido en este documento es gran utilidad en los videojuegos debido a que el mismo otorga realismo y evita se repita el mismo comportamiento monótono una y otra vez.

Sin embargo, el comportamiento emergente que podría tomar el grupo dado un determinado contexto podría no ser el deseado por el diseñador del videojuego en cuestión, ya que dada la gran cantidad de posibilidades se torna muy complejo el testeo y la depuración. Por ejemplo, dado cierta situación de difícil repetición un grupo de soldados podrían dirigirse hacia el lugar equivocado, incluso podrían quedar dando vuelta en círculos. Es por esta razón, que estas técnicas muchas veces son limitadas o aplicadas con reglas muy estudiadas.

Bibliografía

Sitio de Craig Reynolds: <http://www.red3d.com/cwr/boids/>