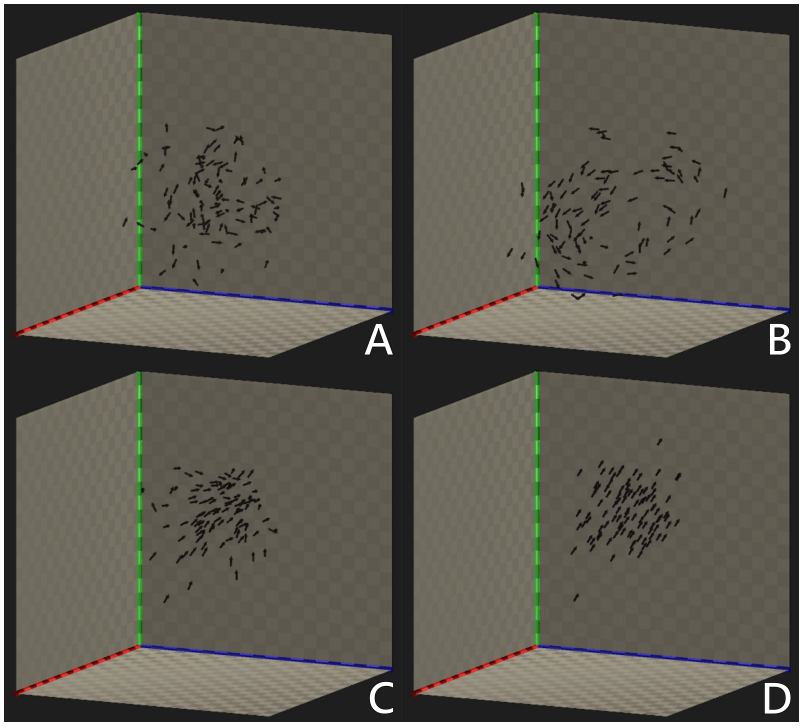
Couzin's Algorithm

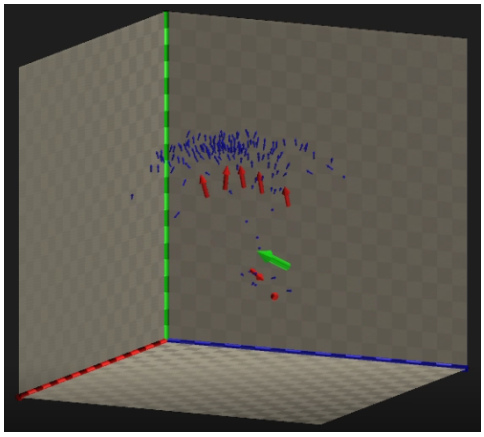
概要

boidのアルゴリズムのように魚や鳥などの群れ行動をシミュレーションします． Couzinのアルゴリズムはboidよりかは少しばかり複雑なアルゴリズムです．

パラメータを変化させることで集団のふるまいが大きく変化します．



複数種族の相互作用を考えることで捕食・逃避行動をシミュレーションできます．



Unityのバージョンは， `2019.4.13f1` で作成しました．それ以外のバージョンでは動作確認していません．

目次

シーンの説明

全部で以下の5つのシーンが用意されています． 好きなシーンを開いて，Unity Editorの画面上部中央のPlayボタンを押せば動かすことができます． 再生中にはGame画面の左上に表示されるTime Scaleのスライダーでプログラムの実行速度を調整できます．

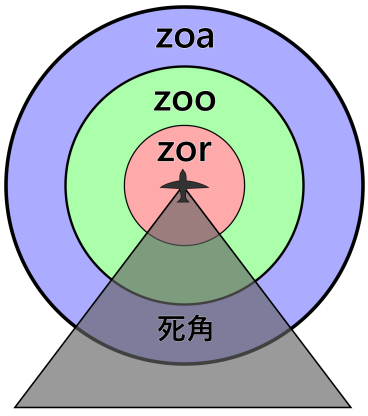
- SingleGroup.unity 1種族のみがランダムに配置されます． パラメータを変化させることで集団のふるまいが変化することが確認できます．
- PreyPredator.unity 捕食者と被捕食者がランダムに配置されます． 捕食・逃避行動のシミュレーションができます．
- PreyPredator2.unity 捕食者と被捕食者が向かい合うようにに配置されます． パラメータによって散開的拡散や湧出効果のような逃避行動が見られます．
- PreyPredator3.unity 捕食者・被捕食者に加え頂点捕食者が配置されます． （このプログラムでは頂点捕食者は底辺被捕食者を捕食しません．）
- PlayGround.unity エージェントや障害物を自由に配置して遊ぶことができます． デフォルトでは捕食関係にない3種族のエージェントが配置されます．

アルゴリズムの説明

Couzinのアルゴリズムでは，各個体が周囲の個体の位置・速度の情報をもとに次に移動する方向を決定します． その基本原理は，個体間で適切な距離を保ちながら，かつ，他の個体から離れすぎないというものです．

詳しい説明は参考文献をご参照ください．

周囲に自らと同種族のみが存在する場合



自らの周囲を距離に応じて3種類の領域に分割します． （下図）

それぞれの領域の意味合いは以下の通りです．

- **Zone of repulsion (zor)** に存在する個体とは距離をとろうとします．
- **Zone of orientation (zoo)** に存在する個体と同じ方向に整列しようとします．
- **Zone of attraction (zoa)** に存在する個体に近づこうとします．

また，zor外の死角にいる個体は感知しません．

各個体の次に移動する方向は以下のルールで決定します．

- zorに別の個体が存在する場合，それらの個体から離れるようにします．これが最優先事項です．
- それ以外の場合，zooの個体とは向きを合わせるようにしながら，zoaの個体には近づくようにします．

それぞれの領域の広さを変化させると，最初の図のように異なった集団のふるまいが見られます． （A: 群集，B: トーラス，C: 動的な併進，D: 高度な併進）

周囲に他の種族の個体が存在する場合

zorや索敵範囲内に他の種族の個体が存在する場合は，相手が獲物であればそれに近づくように，そうでなければ離れるように移動する． それ以外の場合は，上の周囲に自らと同種族のみが存在する場合と同じです．

スクリプトの説明

- CouzinAgent.cs Couzinのアルゴリズムに従うエージェントを実装するスクリプトです． これをオブジェクトにアタッチすることで，独立に動作するエージェントになります． また， `CouzinAgent` クラスの継承元である `Agent` クラスを継承して同様のクラスを自作すれば， Couzinのアルゴリズムをアレンジしたり，それ以外のアルゴリズムを試してみたりすることができます．

- AgentManager.cs 主に，設定されたエージェントを複製してワールドにランダムに配置するためのスクリプトです． 他には，参考文献で紹介されている，集団極性や集団角運動量などの指標を計算する関数も一応実装されています．

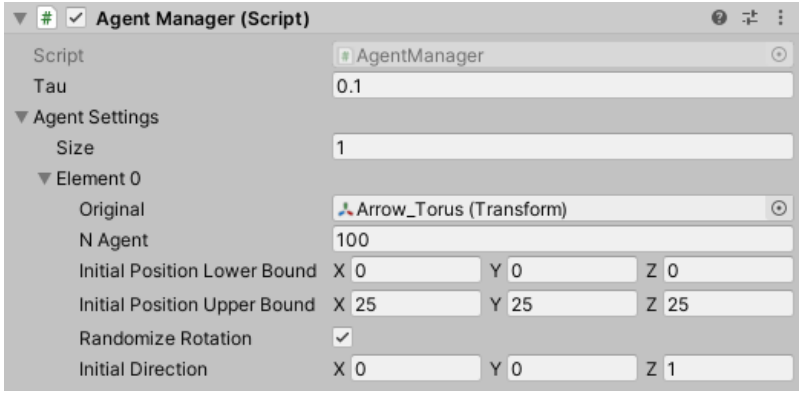
Unity Editor上での操作説明

とりあえず動かしてみる

まず，本プロジェクトをUnity Hubなどから開きます． そして，ProjectタブのAssets > Scenesから適当なシーンを選び開きます． 画面上部中央にあるPlayボタンを押せば開始されます． Play中はGame画面の左上のTime Scaleのスライダーで実行速度を調整できます．

配置されるエージェントを変更する

Playする前に，AgentManagerの設定を変更します． HierarchyタブでManagerを探し，選択します． Inspectorタブで下図のようなAgentManagerの欄を見つけます．



Agent Settings > Element > Originalの欄の右端にある丸を押すと選択ウィンドウが出てきます． ここから，あらかじめ用意されたいくつかの設定のエージェントが選べます．

ついでにAgent Managerの他の設定項目を紹介しておきます． Tauは計算の時間幅[s]です． Agent SettingsのSizeを変更することで複数種類のAgentが設定できるようになります． それぞれのElementの，Originalは複製するエージェントの見本，N Agentは複製する数，その下の項目でランダム配置される場所の範囲や，エージェントが最初に向く方向を指定することができます．

エージェントのパラメータを変更する

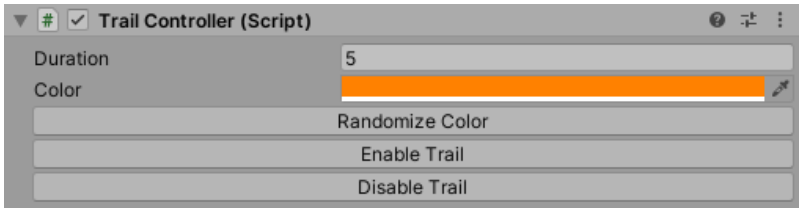
- Playする前であれば， HierarchyタブからManagerで設定したエージェントの見本を探し選択する．
- Play中であれば， 変更したいエージェントをHierarchyタブで見つけ選択する． （複数のエージェントの設定を一括で変更したいとき，検索や複数選択が役に立ちます．）

そして，InspectorタブでCouzin Agentの欄で編集を行う． 各パラメータの説明は，CouzinAgent.cs中のコメントや参考文献をご参照ください．

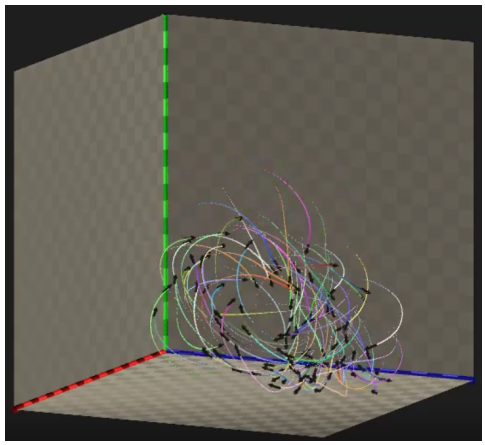
また，エージェントの種族はTagやLayerで管理されています． Inspectorタブの上部で変更が可能です． （注：TagでBlueなどとしてもそれだけでエージェントが青くは残念ながらなりません．InspectorタブでSet Colorの欄を探し，そこで設定をしてください．）

エージェントの軌跡を表示する

Play中に軌跡を表示させたいエージェントをHierarchyタブで見つけ，選択する． （この時，検索，複数選択も便利です．） Trail Controllerの欄（下図）を見つける．



Enable Trailのボタンを押すことで軌跡が表示されます． durationは軌跡の持続時間[s]です． Colorで軌跡の色変更，Randomize Colorでランダムに色を変更します．



矢印以外のモデルを使う

お好きなオブジェクトにCouzinAgent.csとRigidbodyをアタッチすれば，動作すると思います．

障害物を設置する

お好きなオブジェクトを設置した後に，Inspectorタブの上部のLayerをObstacleに設定します．

参考文献

- I. D. Couzin, J. Krause, R. James, G. D. Ruxton, and N. R. Franks, "Collective Memory and Spatial Sorting in Animal Groups," *Journal of Theoretical Biology*, vol. 218, no. 1, pp. 1-11, 2002.
- 伊庭斉志, 深層学習とメタヒューリスティクス ディープ・ニューラルエボリューション, オーム社, 2019.