Tech Design Proposal: *AI Aggressor Management*

Written by Isaiah Everin - Design Portfolio Sample

Overview

For an AI-controlled character to attack another character, they must first claim an “aggressor slot” on that character. This slot is part of a limited pool that is constantly reevaluated based on a variety of prioritization rules (ideally using an existing actor prioritization system) to determine which enemy combatant the player’s attention should focus on.

Aggressor System Architecture

- An AI-controlled character does not decide for itself when it is going to be an aggressor on another character. Instead, each character’s threat component determines which characters that are currently targeting the owner are allowed to be aggressors.
    - This means that an AI’s behavior controller does not have a service managing its aggressor status the way it has a service managing its current target. All aggressor management happens outside of any behavior controller (since it also needs to happen on a playable character).
    - This is similar to but also separate from the attack ticketing system. That system controls the number of simultaneous active attacks, whereas this system pairs with that to first control the number of simultaneous valid attackers, allowing attackers to use knowledge of their current role to inform other behaviors.
    - While some might argue this doesn’t match “character-focused decision making” (i.e. the AI is being told how to consider its role rather than deciding that heuristically using an internal state), from the perspective of a player, that is completely unnoticeable.
        - Instead what they perceive as roles are traded between characters is an illusion that the most appropriate character is taking action against them at the most appropriate time, giving a greater sense of credible intelligence to all enemies.
        - After all, human beings are capable of an awareness of the actions of others in order to determine their own role in a situation—however, for good game design balancing, trying to make a system behave this way by having every individual “think for themselves” to reach these conclusions causes more headaches than it actually solves design problems.

- Aggressor slots are abstract data types. Later parts of this document assumes a breakdown of “Close” and “Distant” aggressors for an Action RPG, but these slots can be used in any context for controlling enemy aggression.

Aggressor Prioritization

- **Aggressor priority rules** are applied differently for ranged/melee combatants and based on whether the target of the aggressor is the player or an AI-controlled character.
    - What this means is that each target may have a prioritization set applied to them that evaluates all aggressors (again, ideally using existing actor prioritization systems that are outside of the scope of this document).
- Examples of scoring conditions for these prioritizers for all aggressor types are below. (These are similar to, but not exactly the same, as the prioritization that must be done when an AI is selecting their current combat target.)
    - **Character Tier Priority:** By default, characters have a base priority that is generally associated with their combat tier (i.e. grunt, normal, elite, and champion). This base priority should be tunable through something like a character data or attribute system.
    - **Range to Target:** Characters who are closer to the target take a higher priority over characters who are farther away.
    - *(Player Target Only)* **Camera Angle from Target:** Characters who are within a viewing angle of the player’s camera take a higher priority over characters who are not on camera.
    - **Is the Target’s Current Target:** If the character is also being targeted by their target, then they take a higher priority (relevant for games where AI might fight other AI-controlled characters, or in a co-op game).
    - **Duration Since Brain was Active:** Hit reactions and crowd-control effects cause an NPC’s brain to pause. After a flat window of ~4 seconds (to reward other characters with being able to deal some damage to them), the longer an NPC is in a paused state, the lower priority they are for aggressor status.

Action RPG Aggressor Types Example

The following sections describe work that would be done across a variety of systems (action selection, movement positioning systems, etc). This documentation does not serve as instructions for specific implementation tasks as those are highly dependent on the architecture of those other systems, so it rather serves as an overview of intent for how a confluence of systems will link together leading to a clear player experience.

Aggressor Types

A given enemy AI character has an **Aggressor Type** property.

- These types are driven by data, with the possibility for designers to create new types.
    - In an Action RPG, these types might be “Close Aggressor” or a “Distant Aggressor” (the examples used throughout this document).
    - As another example, in a First-Person Shooter, the types might be related to more varied behaviors, such as “Flanker,” “Mid-Range,” and “Sniper”.
- This determines which type of slot the character can be considered valid for when a target’s Threat Component is managing aggressor prioritization and slot assignment.
- An enemy can potentially be tuned to occupy multiple types of slots, picking which slot to occupy based on specific tuning and which slots are available. The slot they occupy may then influence other aspects of their behavior.
- Every target has a number of tunable slots per type (i.e. a playable character likely has more slots available than an AI-controlled character).
    - Additionally, the target can define the maximum total number of aggressors independent of type. This is more relevant if there are >2 types of aggressors (such as the first-person shooter example).

Close Aggressor Slot

Per Target: ~3 Close Aggressors (potentially tied to difficulty)
Enemy Types: Melee characters and ranged characters that attack from <=5m away.

Close Aggressor Movement

- **Engage to Attack Movement:** This movement happens when the character is actively attempting to perform an attack against the target. Generally this would be moving into attack range to activate the attack.
- **Passive Combat Movement:** These movement options happen when the character is not actively attacking the target.
    - **Hold Position:** Close Aggressors hold a reasonably close position to the target, a position from which most of their abilities should be able to activate without having to move towards the target first. They also try to stay in the target’s camera rotation (if their target is the player).
    - **Pursue Target:** Close Aggressors pursue to the outer extent of their hold position movement tuning.

    - **Disengage:** Close Aggressors do not disengage from their target unless it's part of an ability. For instance, an assassin archetype might have a disengage dodge ability that will take them far away from the target, at which point they will revoke their aggressor slot (ability systems might potentially allow for forcibly revoking a slot, though it may be sufficient to rely on prioritization in this context).

Distant Aggressor Slot

Per Target: ~2 Distant Aggressors (potentially tied to difficulty)
Enemy Types: Ranged characters that attack from <5m away.

Special Type Priorities

- **Available Abilities:** Ranged characters who have available ranged abilities off of cooldown take a higher slot priority. The goal here is to make it so that a character does not retain a Distant Aggressor slot if they have no available actions to take. (Remember that this is multiplied against many other factors, it does not guarantee the ranged role will constantly cycle between opponents.)

Distant Aggressor Movement

- **Engage to Attack Movement:** This movement happens when the character is actively attempting to perform an ability against the target. For Distant Aggressors, this will only be used when the target is out of range of their abilities.

- **Passive Combat Movement:** These movement options are performed the same for a Distant Aggressor as they are for an Inactive Aggressor.

Inactive Aggressors

Inactive Aggressor Action Selection

- Characters who haven't been assigned aggressor slots do not perform normal attacks, but they may still perform reactive actions (such as signal-driven dodges).

- Inactive Close Aggressors are allowed to perform "Opportunity Attacks." These attacks must be flagged for use, and they are able to steal current reserved (but not active) attack tickets as well as aggressor status.

    - This is all assuming an attack ticket system where attackers reserve tickets while preparing to perform the attack in advance (i.e. they are now moving towards the target). It allows for an inactive aggressor to become active and begin attacking, forcing the attackers who are currently engaging to revoke the attack they were preparing for.

    - For example, a player would see Enemy A running towards them to perform a sword swing. The player backs up, and sees they are getting within range of Enemy B who is now much closer to them than Enemy A. At that moment, Enemy B selects an opportunity attack, taking the tickets and slot from Enemy A and starting a melee attack against the nearby player. Enemy A returns to passive movement (see below).

    - It is important to be able to flag which attacks can activate in this context, as some very powerful attacks might be inappropriate to suddenly trigger with this reactive juggling of the current aggressor.

Inactive Aggressor Movement

- **Engage to Attack Movement:** Inactive Aggressors do not engage directly with their target. They only pursue targets (see below).

- **Passive Combat Movement:** Inactive Aggressors do not attack their target, so these are the core behaviors that Inactive Aggressors perform.

    - **Hold Position:** Inactive Aggressors try to hold a relatively distant position to their target, giving the Close Aggressors room to engage the player. Inactive Aggressors tagged as “Close Aggresor Type” will hold a slightly closer position, Inactive Aggressors tagged as “Distant Aggressor Type” will hold the same position as Distant Aggressors. Inactive Aggressors do not move with concern for the player’s camera rotation.

    - **Pursue Target:** Inactive Aggressors will pursue to the outer extent of their hold position movement patterns.

    - **Disengage:** When a character becomes an Inactive Aggressor, they will disengage to the inner extent of their hold position movement patterns.