

А. А. Лаптев, Н. Б. Осипенко
(ГГУ им. Ф. Скорины, Гомель)
**ПРОДВИНУТЫЙ ИГРОВОЙ ИНТЕЛЛЕКТ
ДЛЯ ИНТЕРАКТИВНОГО ВЗАИМОДЕЙСТВИЯ
С ПОЛЬЗОВАТЕЛЕМ.**

В работе описан продвинутой игровой интеллект, реализованный в игровой среде Unity3D и на языке C#, компоненты которого могут использоваться начинающими программистами для изучения и создания своих игр.

Продвинутой игровой интеллект (ИИ) умеет двигаться к цели и атаковать врага с определенной задержкой, преследовать цель, потерянную из зоны видимости, обходить и разрушать препятствия. Представление об алгоритме работы улучшенного ИИ дает рисунок 1.



Рисунок 1 – Схема работы продвинутого игрового интеллекта

Из схемы видны возможные переходы между семью исходными состояниями ИИ: бездействие (ИИ стоит на месте и просматривает, не появился ли кто-либо в зоне видимости); тревога (ИИ замечает в зоне видимости кого-либо); движение к цели (активируется, когда игрок входит в зону для преследования ИИ и тот начинает двигаться к игроку); обход препятствия (ИИ обходит препятствия, которые попадают на пути преследования); разрушение препятствия (ИИ разрушает препятствие, которое нельзя обойти и которое можно разрушить); продолжение движения к цели (при выходе из зоны видимости ИИ двигается еще некоторое время в последнюю точку, в которой был игрок); атака (после достижения цели, подойдя на определенное допустимое расстояние, ИИ начинает атаковать игрока).

Алгоритм работы продвинутого ИИ реализован в классе `AIImproved.js`, его основные атрибуты выполняют следующие функции: `state_stand` – состояние бездействия; `state_walk` – состояние движения; `state_attack` – состояние атаки; `condition_see_enemy` – условие видимости врага; `condition_can_attack` – условие атаки врага, `condition_can_stand` – условие бездействия, `condition_can_walk` – условие движения; `condition_can_obstacle` – условие наличия препятствия; `update()` – обдумывание действий, выбор обстановки, получение текущего набора действий; `selectNewSchedule(condition:Array)` – исполнение текущего набора действий; `getCondition()` – генерация и возврат новых условий; `onInitStand()` – установление ожидания нового действия; `onStand()` – переключение на новое действие; `onInitWalk()` – установка решений по движению; `onWalk()` – движение.

Данное приложение демонстрирует продвинутые возможности игровых интеллектов, а функциональность приложения может быть доработана путем добавления дополнительных обчетов ситуаций и настроек для более гибкого управления игровым интеллектом.