

ФОРМИРОВАНИЕ ПОВЕДЕНИЯ ИГРОВОГО ИИ: ИСПОЛЬЗОВАНИЕ МЕТОДОВ МАШИННОГО ОБУЧЕНИЯ С ПОДКРЕПЛЕНИЕМ ДЛЯ АВТОМАТИЧЕСКОЙ ГЕНЕРАЦИИ ИЕРАРХИЧЕСКИХ СЕТЕЙ ЗАДАЧ

Костюк Д.С. (ИТМО)

Научный руководитель – Трифанов А.И. (ИТМО)

Введение. Современная индустрия компьютерных игр активно использует интеллектуальные системы для управления поведением неигровых персонажей. Поведение игрового агента определяется набором правил, логики и алгоритмов, направленных на достижение целей в динамической виртуальной среде. Среди наиболее распространённых подходов применяются конечные автоматы, деревья поведения, системы на основе правил и методы планирования, включая иерархические сети задач (HTN).

HTN представляют собой удобный инструмент для моделирования сложных стратегий поведения за счёт декомпозиции глобальной цели на последовательность подзадач. Однако их проектирование традиционно выполняется вручную и требует значительных временных затрат, а также экспертных знаний. Это ограничивает их применение в крупных игровых проектах с высокой вариативностью поведения агентов.

Одним из перспективных направлений автоматизации разработки игрового ИИ является использование методов обучения с подкреплением. Такие методы позволяют агенту самостоятельно формировать стратегии поведения на основе взаимодействия со средой и накопленного опыта. В связи с этим актуальной является задача автоматической генерации структуры HTN на основе данных, полученных в процессе обучения агента.

Основная часть. Иерархическая сеть задач (HTN) – это метод планирования в области искусственного интеллекта, который используется для решения сложных задач путем декомпозиции их на более простые подзадачи. HTN принимает на вход задачу и предоставляет на выходе серию шагов, которые ее решают. В терминах иерархических сетей задач серия шагов называется планом. Основными элементами HTN являются:

- 1) Задачи (Tasks) – основные цели, которые необходимо достичь;
- 2) Методы (Methods) – способы декомпозиции задач на подзадачи;
- 3) Операторы (Operators) – конечные действия, которые может выполнить агент;
- 4) Сеть задач (Tasks network) – график, представляющий иерархическую структуру задач и подзадач, иллюстрирующий взаимосвязи и зависимости между ними;
- 5) План (Plan) – последовательность операторов и подзадач, которые приводят к решению поставленной задачи.

С того момента как в 1975 году [1] было предложено планирование HTN, оно применялось во многих областях, включая:

- 1) Разработку планов действий правительственными ведомствами;
- 2) Управление интеллектуальными роботами;
- 3) Управление поведением персонажей в шутерах от первого лица (FPS) и стратегиях в реальном времени (RTS) в играх.

Целью работы является разработка подхода к автоматической генерации иерархических сетей задач для игрового ИИ на основе анализа данных, полученных в процессе обучения с подкреплением.

В рамках исследования был разработан игровой сценарий в среде Unity, моделирующий соревновательную задачу между командами агентов. Игровое поле содержит ресурсы, за которые команды получают очки, а агенты способны перемещаться, взаимодействовать с объектами и вступать в бой с противниками.

На начальном этапе были реализованы вручную модели поведения на основе HTN и rule-based для тестирования среды и базовой логики взаимодействия. Далее для обучения агентов применялся инструментарий Unity ML-Agents с использованием алгоритма обучения с

подкреплением PPO [2]. Агенту был задан фиксированный набор действий и наблюдений, а система вознаграждений была сформирована таким образом, чтобы стимулировать достижение игровой цели – победу над противником.

Обучение проводилось в параллельных игровых средах, что позволило ускорить процесс накопления опыта. В результате агент научился демонстрировать устойчивую стратегию поведения, обеспечивающую получение вознаграждения и достижение целей в игровой среде.

После завершения обучения были собраны логи работы модели, содержащие информацию о принимаемых решениях, состоянии среды и накопленных наградах. На основе анализа этих данных был разработан метод извлечения структуры поведения агента.

В ходе анализа:

- Выделялись элементарные действия агента как базовые задачи;
- Определялись наиболее значимые действия на основе частоты использования и полезности, этот метод основан на предыдущем исследовании [3];
- Формировалась иерархия задач с учётом важности действий в различных ситуациях;
- Определялись типичные условия выполнения действий и их последствия.

На основе полученной структуры автоматически генерировались программные модули, реализующие HTN в среде Unity. Таким образом, сформированная иерархия задач представляла собой интерпретацию стратегии поведения обученного агента.

Экспериментальная проверка показала, что сгенерированная HTN-модель воспроизводит ключевые поведенческие паттерны, сформированные в процессе обучения с подкреплением, и демонстрирует сопоставимую эффективность при взаимодействии с агентами, использующими заранее заданные модели поведения.

Выводы. В работе предложен подход к автоматической генерации иерархических сетей задач на основе анализа поведения агента, обученного с подкреплением в игровой среде.

Полученные результаты показывают, что:

- Обучение с подкреплением позволяет сформировать устойчивую стратегию поведения агента;
- Данные, полученные в процессе обучения, могут быть интерпретированы в структурированную модель HTN;
- Автоматически сгенерированная HTN способна воспроизводить характерные решения обученной модели и демонстрировать сопоставимую эффективность.

Практическая значимость работы заключается в снижении трудозатрат на проектирование поведения игрового ИИ и возможности автоматизированного переноса полученных стратегий в структурированные модели планирования. Предложенный подход может быть использован при разработке неигровых персонажей и исследовании адаптивных систем поведения в видеоиграх.

Список использованных источников:

1. Sacerdoti E.D. The nonlinear nature of plans // Stanford Research Inst. – 1975. – 100 p.
2. Schulman J., Wolski F., Dhariwal P., Radford A., Klimov O. Proximal Policy Optimization Algorithms [Электронный ресурс] // arxiv.org. – 2017. – Режим доступа: <https://arxiv.org/abs/1707.06347>, свободный. – (Дата обращения: 05.11.2025).
3. Dworzanski R., Hlavacs H. Shaping AI behavior: a Q-learning driven approach to automatic behavior tree creation [Электронный ресурс] // University of Vienna. – 2023. – Режим доступа: <https://theses.univie.ac.at/detail/68504>, свободный. – (Дата обращения: 12.02.2025).