Планирование на основе GOAP в задаче поиска объекта группой мобильных роботов

Алексей Линявский [1,2] и Надежда Нагул [2]

[1] *Иркутский государственный университет путей сообщения, ул. Чернышевского, 15, Иркутск, 664074, Россия*
[2] *ИДСТУ СО РАН им. В.М. Матросова, ул. Лермонтова, 134, 664033, Иркутск, Россия*

Аннотация

В докладе рассматривается подход к автоматическому планированию поведения автономного мобильного робота на основе архитектуры GOAP (Goal-Oriented Action Planning). Показаны примеры формализации возможных действий роботов для применения GOAP, приведены примеры получаемых планов для решения задачи поиска и исследования объектов, расположенных в некоторой области с препятствиями.

Ключевые слова

Автономные мобильные роботы, автоматическое планирование, принятие решений

1. Введение

Goal-Oriented Action Planning – широко применяемая в компьютерных играх архитектура планирования действий так называемых NPC (Non-Player Character), позволяющая достичь видимости естественного интеллектуального поведения персонажей [1, 2]. GOAP был представлен в 2005 году как модификация планировщика STRIPS (Stanford Research Institute Problem Solver [3]), направленная на ускорение построения планов в динамических средах и их упрощение для более эффективной реализации [4]. Как и в других планировщиках, суть GOAP состоит в составлении плана достижения выбранной цели из доступного в текущей ситуации ограниченного набора действий, имеющих условия для их выполнения и эффекты, наступающие после выполнения. В отличие от STRIPS, каждому действию была назначена стоимость, которая использовалась в качестве эвристики для алгоритма A* для поиска в пространстве состояний мира. Кроме того, GOAP использует процедурные предусловия, с помощью которых система в процессе выполнения плана может использовать более сложную логику, например, вызывать функции планирования пути или распознавания объекта.

Преимуществом GOAP является возможность реализации сложных поведений объектов, для которых неприменимы традиционные алгоритмы планирования. Известным подходом к планированию и принятию решений является описание задач в языке PDDL и применение специализированных солверов. Однако разработка PDDL-планировщика представляет собой трудозатратную работу, при этом его функции могут оказаться невостребованными в решаемых задачах. В докладе рассматривается возможность использования подхода на основе GOAP для управления автономными мобильными роботами в процессе исследования заданной области и некоторых объектов в ней. В то время как для решения части задач, составляющих миссию роботов, например, для планирования пути и назначения заданий, используются известные алгоритмы [5, 6], принятие решений о той или иной последовательности высокоуровневых действий осуществляется с помощью GOAP.

7th International Workshop on Information, Computation, and Control Systems for Distributed Environments (ICCS-DE 2025), July 7–11, 2025, Irkutsk, Russia
EMAIL: inetlsec@mail.ru (A. 1); sapling@icc.ru (A. 2)
ORCID: 0009-0008-0412-9158 (A. 1); 0000-0003-1439-3274 (A. 2)

ICCS-DE 2025 Workshop Proceedings
DOI: 10.47350/ICCS-DE.2025.30

2. Goal-Oriented Action Planning

В качестве входных данных планировщик GOAP принимает: множество агентов, множество возможных действий каждого агента, множество задач, начальные состояния и цели. Цели представляют собой состояния, которое агент должен достичь из начального или текущего состояния (рис. 1). В общем случае, состояния равны тогда и только тогда, когда наборы параметров состояний совпадают, а также совпадают значения параметров внутри наборов.

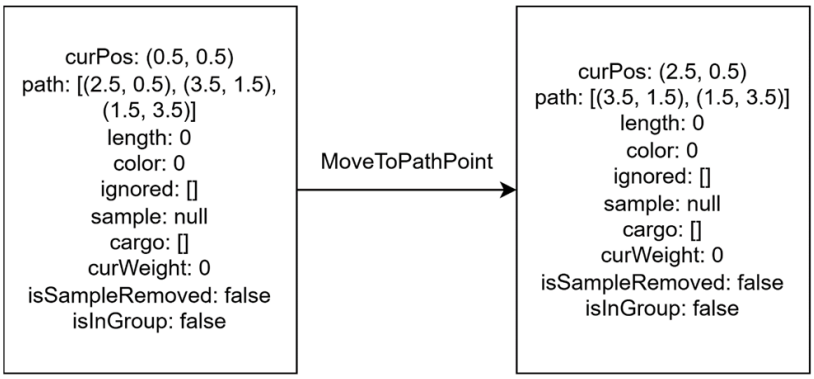

Рисунок 1: Пример состояний робота. Каждая строка представляет собой некоторый параметр.

Каждое действие в GOAP описывается следующими компонентами (рис. 2):

- вес действия - используется для оценки релевантности действия в момент планирования;
- предусловия - условия, необходимые для выполнения действия;
- эффекты - состояние, которое агент принимает после выполнения действия;
- алгоритм реализации действия.

Рисунок 2: Описание действия в GOAP

Представление в GAOP предварительных условий и эффектов действий в виде массивов переменных состояния системы фиксированного размера позволяет быстрее найти действие, которое могло бы уменьшить разницу между начальным и целевым состоянием посредством прямого сравнения массивов. План из действий составляется таким образом, чтобы последовательность действий привела начальное состояние к конечному - цели. Для построения плана в GOAP, как и STRIPS, используется обратное планирование, так называемое планирование от цели [1].

3. Задача исследования объектов группой AMP

Рассмотрим следующую миссию для группы из трех автономных мобильных роботов (AMP). Пусть задана некоторая область, и построены маршруты движения каждого из AMP для совместного сканирования области в поисках мест скопления некоторых предметов, возможно, представляющих интерес для дальнейшего исследования. При обнаружении скопления образцов робот может отделиться от группы для самостоятельного их исследования, либо вся группа проводит исследование совместно. Приведем несколько возможных действий каждого AMP группы:

1. **Движение по пути группой:** группа AMP движется к точке point в path.
   GroupAction: MoveToPathPointGrouped(point)

*Weight: dist(curPos, point) / (10 * (isInGroup ? 1.5 : 1))*
Preconditions:
isInGroup = true
!path.isEmpty()
pathExists(curPos, point)
!ingored.contains(point)
Effects:
curPos = point

2. **Выход из группы:** AMP перестаёт двигаться в группе и начинает работать отдельно от группы из точки point.

Action: LeaveGroup(point)
Weight: (isInGroup ? 0.25 : 1)
Preconditions:
isInGroup = true
curPos = point
Effects:
isInGroup = false

3. **Исследование образца:** AMP выполняет обследование образца в точке point.

Action: ScanSample(point)
Weight: 1
Preconditions:
isInGroup = false
color = 0
length = 0
Effects:
color != 0
length != 0
sample = getSampleByColorLength(color, length)

4. **Игнорирование образца:** AMP записывает образец sample в игнорируемые ignored.

Action: Ignore()
Weight: getWeight(sample)
Preconditions:
sample = null
Effects:
isSampleRemoved = true
ignored += [curPos]

5. **Поднятие образца:** AMP собирает образец sample в свой грузовой отсек.

Action: TakeSample(sample)
*Weight: getWeight(sample) * curWeight*
Preconditions:
sample != null
curWeight + getWeightCost(sample) <= 1
Effects:
cargo += [sample]
curWeight += getWeightCost(sample)
sample = null

В зависимости от текущего состояния системы, присвоенные действиям веса влияют на выбор одной из возможных последовательностей действий, формируя таким образом план. Для построения плана используется алгоритм A*. На рис. 3 цветом выделен реализуемый план, выбранный по параметрам исследованного образца и веса каждого из действий.

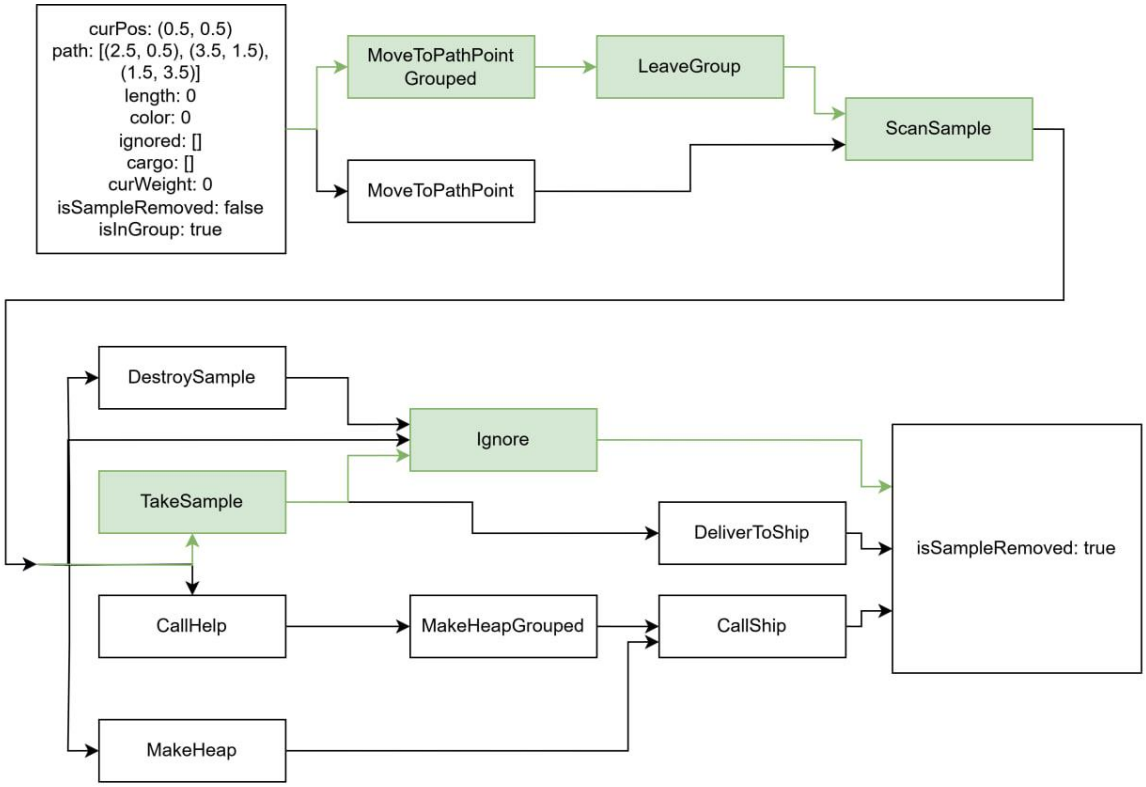

Рисунок 3: Возможные планы, полученные GOAP для группы AMP. Выбранный план помечен зеленым цветом.

4. Заключение

В докладе описаны принципы использования GOAP для планирования высокоуровневых действий мобильных роботов. Построение плана проводилось для индивидуального робота, однако в дальнейшем будут исследованы возможности применения GAOP с кооперирующимися агентами [7], а также рассмотрены более сложные подходы к планированию, такие как HTN (Hierarchical Task Network), в котором используются принципы иерархии задач и их декомпозиция [4]. Описанная в докладе миссия будет реализована в робототехническом комплексе TEMAR для тестирования предложенного подхода.

Список литературы

[1] J. Orkin. Applying goal-oriented action planning to games. AI Game Programming Wisdom, vol. 2, no. 2004, pp. 217–227, 2004.

[2] M.Ç. Uludağlı, K. Oğuz. Non-player character decision-making in computer games. Artif Intell Rev 56, 14159–14191 (2023). https://doi.org/10.1007/s10462-023-10491-7

[3] R. E. Fikes, N. J. Nilsson. STRIPS: A new approach to the application of theorem proving to problem solving, Artificial intelligence, vol. 2, no. 3-4, pp. 189–208, 1971.

[4] X. Neufeld, S. Mostaghim, D. L. Sancho-Pradel and S. Brand. Building a Planner: A Survey of Planning Systems Used in Commercial Video Games. IEEE Transactions on Games, 11, 2 (2019), 91-108. DOI: 10.1109/TG.2017.2782846.

[5] J. Li, Y. Zhao, Y. Shen. Multi-Factor Task Assignment and Adaptive Window Enhanced Conflict-Based Search: Multi-Agent Task Assignment and Path Planning for a Smart Factoryб Electronics, 14, 5 (2025), 842. DOI: https://doi.org/10.3390/electronics14050842.

[6] J. Tang, H. Ma. Mixed Integer Programming for Time-Optimal Multi-Robot Coverage Path Planning with Efficient Heuristics, Proceedings of the International Symposium on Combinatorial Search, 17 (2024), 289-290. DOI: https://doi.org/10.1609/socs.v17i1.31585.
[7] G. Boeda. Multi-agent cooperation in games with goal oriented action planner: use case in WONDER prototype project, Proceedings of the AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment. 17, 1 (2021), 204-207. DOI: https://doi.org/10.1609/aiide.v17i1.18909.