

Санкт-Петербургский государственный университет

Системное программирование

Группа 22.М07-мм

Прототип средства моделирования для сравнения стратегий управления группой роботов

Ершов Владислав Евгеньевич

Отчёт по учебной практике
в форме «Решение»

Научный руководитель:
профессор кафедры системного программирования, д.ф.-м.н. О. Н. Граничин

Санкт-Петербург
2022

Saint Petersburg State University

Advisor's chair

Group 22.M07-mm

Vladislav Ershov

Implementing the Clustering Algorithm in the Simulator

Internship report
in a «Solution» form

Scientific supervisor:
Sc.D, director of the Scientific and Educational Center of St. Petersburg State University
"Mathematical Robotics and Artificial Intelligence" K.S. Amelin

Saint Petersburg
2022

Оглавление

Введение	4
Постановка задачи	6
1. Обзор	7
1.1. Мультиагентный решения	7
1.2. Средства моделирования роботов	8
2. Метод	9
2.1. Прототип	9
2.2. Мир симуляции	9
3. Реализация	11
3.1. Стратегия микро-управления	11
3.2. Стратегия макро-управления	11
4. Эксперимент	12
4.1. Описание стратегий	12
4.2. Метрики	12
4.3. Результаты	12
4.4. Обсуждение результатов	13
Заключение	14
Список литературы	16

Введение

За последние десять лет было проведено много исследований в сфере разработки и анализа алгоритмов управления мультиагентными системами. Рост интереса к мультиагентной парадигме обусловлен новыми идеями ее применения. Например, такую парадигму можно использовать для групп беспилотных автомобилей, транспортных систем, распределенных сенсорных сетей, интернета вещей и много другого [Ren and Beard (2008) Xia et al. (2022)]. Таким образом, за последнее время появилось огромное число различных стратегий управления для мультиагентных систем, которые эффективно работают [Pasqualetti et al. (2014 г.); Баджо и др. (2021) и ссылки в нем].

Однако, большинство таких стратегий требуют корректировку в реальном времени в соответствии с текущей ситуацией. Поэтому быстро сходящиеся стратегии управления становятся все более востребованными.

Существует два основных класса стратегий управления, а именно макро-управление и микро-управление. Макро-управление является наиболее простым способом – каждый узел системы управляется извне и получает одинаковые для всех узлов управляющие воздействия. Микро-управление обеспечивает наиболее точное решение – каждый узел системы управляется индивидуально. Оба класса стратегий управления имеют ряд недостатков.

В качестве примера можно рассмотреть группу беспилотных роботов, развернутых на большом расстоянии от центра управления. Удаленно управлять каждым вездеходом в режиме реального времени нецелесообразно, так как сигналы управления будут поступать с задержками. Для данного примера иерархическое управление выглядит более перспективным решением.

В иерархическом управлении первый уровень состоит из узлов в группе, которые следуют за своим лидером группы. На втором уровне лидеры групп становятся узлами, которые, в свою очередь, имеют своих лидеров. Аналогичное разбиение справедливо для большего количе-

ства уровней. Иерархическое управление позволяет увеличить скорость конвергенции (схождения) [Mukherjee and Ghose (2016)].

В отличие от этих стратегий управления, наша статья фокусируется на новом – мезо-управлении. Мезо-управление частично наследует преимущества иерархического контроля; однако для этого не требуется априорного знания структуры системы. Вместо этого он опирается на самоорганизующееся поведение сложных систем, что позволяет моделировать адаптивную динамику для сложных роботизированных сетей, которые стремятся работать в среде с возмущениями и неопределенностями. Недавние достижения [Проскурников и Граничин (2018), Граничин и Ужва (2022)] показывают, что роботизированные системы со сложной динамикой и взаимодействиями демонстрируют так называемую кластерную синхронизацию, также называемую кластеризацией. Согласно исследованиям, кластерная синхронизация в основном возникает в системах с неполной связностью между агентами и из-за внешних возмущений, которые на самом деле могут также влиять на связность и состояния агентов, помимо других факторов, приводящих к кластеризации.

Эффективный кластерный контроль описан в [Wang et al. (2019), Юань и др. (2019) и Граничин и Ужва (2021)], последние два в основном сосредоточены на беспроводных сенсорных сетях. С целью рассмотрения возможных приложений к разреженным представлениям коммуникационных сигналов была предложена другая родственная работа [Граничина и Ужвы (2022), в которой методология сжатого восприятия (см. Candes et al. (2006) Donoho (2006))] используется для синтеза управляющего воздействия в сжатое пространство представления полной системы.

Постановка задачи

Целью работы является разработка и реализация прототипа симуляции, который позволил бы наглядно сравнить работу стратегий микро-, мезо- и макро-управления.

Поставленная цель обусловила решение следующих задач.

1. Сделать обзор средств симуляции для групп роботов.
2. На основе обзора разработать мир симуляции.
3. Реализовать симуляцию.
4. Реализовать алгоритмы микро-, мезо-, макро-управления.
5. Провести практический эксперимент для оценки эффективности стратегий управления.

1. Обзор

В обзоре будем рассматривать средства симуляции.

1.1. Мультиагентный решения

1.1.1. Технология JADE

JADE [1] – программное обеспечение, находящиеся на промежуточном слое. Предназначено для реализации распределенных мультиагентных систем. Есть возможность работать с узлами имеющими ограниченные ресурсы.

Технология состоит из динамической среды, набора графических утилит, библиотеки классов.

JADE включает в себя набор контейнеров, являющихся динамической средой исполнения. При этом, в каждом контейнере находится несколько агентов. Стоит отметить, что у системы есть главный контейнер. Он содержит специальных агентов: AMS – это система управления агентами и DF – сервис «желтых страниц», без данных агентов система не может работать.

JADE является довольно тяжеловесным средством – она позволяет создать систему с огромным количеством узлов, объединяя при этом вычислительные узлы, которые могут иметь абсолютно различную архитектуру и физические возможности, в огромную сеть. Однако это требует довольно больших накладных расходов.

1.1.2. Среда разработки SPADE

SPADE [?] – интеллектуальная среда разработки агента Python. Это платформа для реализации мультиагентных систем, которая написана на Python и основана на обмене мгновенными сообщениями – XMPP.

Предоставляет модель агента, которая состоит из механизма подключения к платформе, набора различных вариантов поведения и диспетчера сообщений. Диспетчер отвечает за передачу сообщений агентам. У каждого агента должен быть идентификатор Jabber ID (JID), а

также пароль для соединения с сервером XMPP.

SPADE обеспечивает возможность коммуникации с помощью протокола XMPP. Протокол предоставляет механизм для регистрации и аутентификации на сервере XMPP.

Каждый зарегистрированный агент имеет открытую и постоянную связь с XMPP сервером.

Однако, такая связь не является непрерывной, то есть агенты могут теряться на не малое количество времени. Причем, чем агентов больше, тем сильнее это сказывается на всей системе.

Данная особенность может довольно сильно сказываться на чистоте экспериментов.

1.2. Средства моделирования роботов

Будем рассматривать средства для более точного симулирования физических законов реального мира, а также более качественного визуализирования результатов симуляции.

1.2.1. Webots

Webots [?] – это кроссплатформенная система моделирования, которая имеет множество интеграций: MATLAB, ROS, API для языков программирования Java, Python, C++.

Система дает возможность использовать богатую библиотеку сенсоров, моделей и агрегатов для роботов; интерактивную визуализацию в 3D; набор уже готовых роботов и окружений, а также инструменты для реализации связи между роботами.

Webots предоставляет пользователю наиболее популярные сенсоры и исполнительные устройства: датчики света, инфракрасные дальномеры, акселерометры, камеры и многие другие.

Благодаря этому можно реализовать и провести моделирование с огромным набором параметров и с учетом физических законов реального мира.

2. Метод

2.1. Прототип

Прототип симуляции должен отвечать нескольким параметрам:

- Быть достаточно простым и абстрактным, чтобы к нему можно было свести большой круг задач.
- Обеспечивать хорошую наглядность.
- Повторять механику взаимодействия с агентами более продвинутых средств симуляции.

2.2. Мир симуляции

Для обеспечения простоты симуляции взята бесконечная плоскость разбитая на шестиугольники правильной формы. При этом, у каждой шестиугольной клетки есть координаты – номер столбца и строки, в которой находится клетка. Такой вариант пространства довольно прост и абстрактен. Также он обеспечивает хорошую наглядность.

Время в мире симуляции идет дискретно, такой подход позволяет избежать не маленьких трудностей. Ведь для симуляции реального времени необходимо реализовывать тяжеловесный механизм, который будет отвечать за него.

Состоянием агента является его расположение на бесконечной плоскости, то есть координатой шестиугольной ячейки, в которой он находится.

Введенные правила мира позволяют более гибко его использовать:

- На одной клетке в один момент времени может находиться только один агент.
- В мире есть клетки-препятствия – на них не может находиться ни один агент.

- В качестве дополнительного параметра влияния на симуляцию вводятся штрафы для агентов, которые нарушают правила мира.
- Агент может получить два вида штрафа – некоторое количество шагов, при котором агент не может передвигаться, но может общаться, либо полное отключение на некоторое количество шагов
- Столкновение агентов – попытка двух и более агентов переместиться в одну клетку. Наказание – все такие агенты остаются на месте, в зависимости от правил конкретного варианта могут получить некоторое количество штрафных шагов

Задачей агента является приведение его состояния к определенному значению, то есть достижение определенной клетки на плоскости.

3. Реализация

Реализация модели симуляции схожа с моделью SPADE. Есть мир, который отсчитывает время и запускает процессы у агентов – передачу сообщений и передвижения агентов. Также он следит за соблюдением агентами правил мира.

3.1. Стратегия микро-управления

В данной стратегии у каждого агента есть индивидуальное управляющее воздействие. Что с одной стороны позволяет агентам более точно исполнять задачу, однако требует большого числа вычислений.

3.2. Стратегия макро-управления

Стратегия макро-управления требует гораздо меньшего количества вычислений для определения пути исполнения задачи. Так как для каждого агента применяется одинаковый усредненный путь исполнения задачи. Однако это приводит к меньшей точности исполнения задачи.

4. Эксперимент

Сравним работу стратегий микро-, мезо- и макроо-управления. Для этого проведем эксперименты с препятствиями и без них.

4.1. Описание стратегий

Опишем стратегии управления для экспериментов.

- Стратегия микро-управления – агенты движутся к цели, избегая препятствий, при этом агенты могут столкнуться. За каждое столкновение агенты получают 10 штрафных шагов.
- Стратегия макро-управления – агенты находят общий центр масс, используя локальный протокол голосования, вычисляют траекторию от центра масс к цели и перемещаются параллельно ей.
- Стратегия мезо-управления – агенты движутся к цели, но группами, которые они сами определяют. Если агент окажется рядом с препятствием, то он может внести небольшое искажение в общий контроль кластера, чтобы обойти препятствие.

4.2. Метрики

Для числового и визуального сравнения введем метрики.

- Метрика точности выполнения задачи – точность группы.
- Метрика групповой точности – диаметр.

4.3. Результаты

При моделировании эксперименты проводились для 40 агентов на бесконечной плоскости.

Рисунок 1 показывает, что мезо-управление более эффективно в динамике достижения цели. Тем не менее, конечный результат немного лучше у микро-управления.

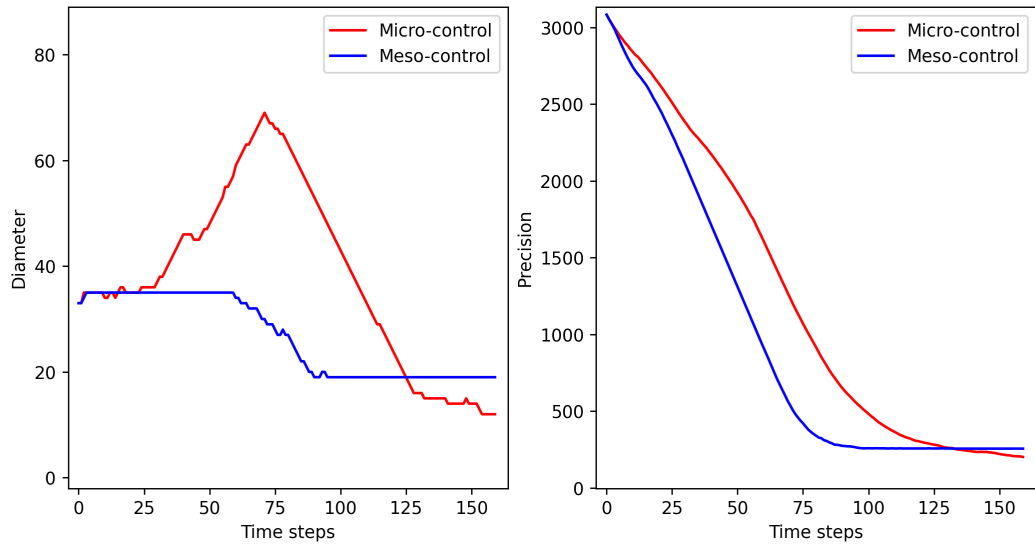


Рис. 1: Измерения диаметра и точности для 40 агентов

4.4. Обсуждение результатов

Для лучшего понимания экспериментов, необходимо масштабировать их результаты.

Нетрудно представить, что у группы будет не одна цель, а несколько, и достигать их надо последовательно. В этом случае лучшая динамика стратегии мезо-управления даст огромное преимущество, ведь чем быстрее достигается одна цель, тем быстрее можно перейти к следующей.

Также, как видно из Рисунок 2 и Рисунка 3 движения агентов, микро-управление очень сильно растягивает группу, что приводит к ухудшению групповой ценности. Это, в свою очередь, приводит к большей разобщенности агентов, что не лучшим образом сказывается на эффективности задачи в целом.

Таким образом, можно сделать вывод, что при усложнении задачи для группы стратегия мезо-управления будет давать еще больший выигрыш по сравнению со стратегией микро-управления.

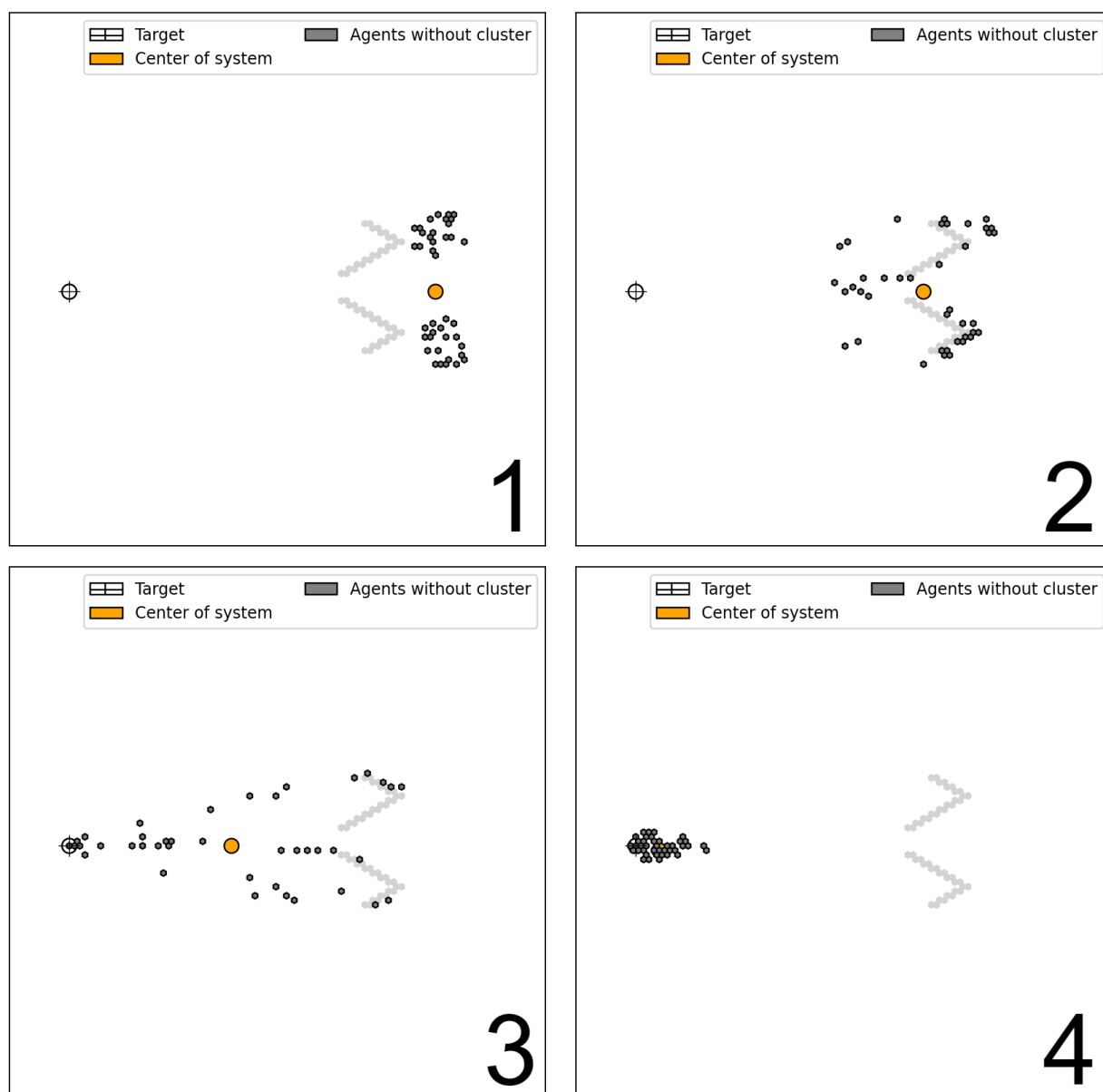


Рис. 2: Динамика агентов со стратегией микро-управления

Заключение

Были достигнуты следующие результаты.

1. Проведен обзор существующих решений в сфере средств моделирования и алгоритмо кластеризации.
2. Реализация алгоритма в в средстве моделирования.
3. Проведен эксперимент в средстве моделирования, который показал, что реализованный алгоритм лучше других алгоритмов.

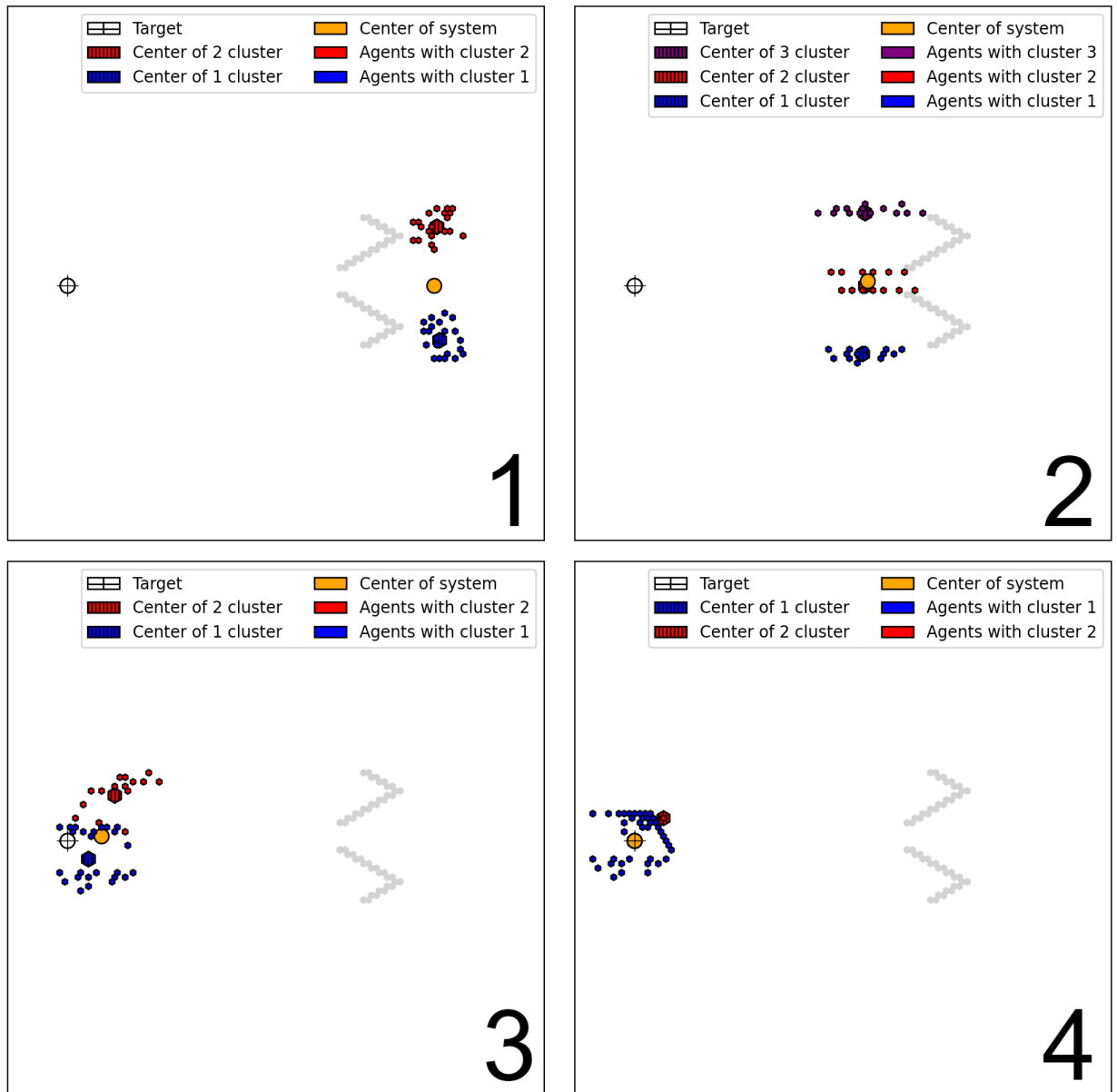


Рис. 3: Динамика агентов со стратегией макро-управления

Код проекта доступен на сайте Github: <https://github.com/ErshovVladislav1/dissertation>

Список литературы

- [1] JADE. — 2010. — URL: <https://techweirdo.wordpress.com/2010/12/01/p73/> (online; accessed: 2022-05-24).