

УДК 004.021

DOI: 10.24412/2071-6168-2025-6-70-71

## ИМИТАЦИОННОЕ МОДЕЛИРОВАНИЕ ДЕЦЕНТРАЛИЗОВАННОГО РОЯ ДРОНОВ ДЛЯ ОПТИМИЗАЦИИ ПОИСКОВЫХ ОПЕРАЦИЙ НА НЕИЗУЧЕННОЙ МЕСТНОСТИ

С. Н. Бориско, И. В. Бородин, А. А. Коротков, А. А. Тимошкин

*Представлен гибридный алгоритм управления роем дронов, объединяющий случайное блуждание с коррекцией на тепловую карту. Цель – решить проблемы классических методов: застревание в локальных минимумах, дублирование маршрутов и неравномерное покрытие зоны поиска. Для синхронизации данных между дронами использованы блокчейн (децентрализация, защита от подделки) и CRDT (бесконфликтная синхронизация). Моделирование выполнено на Python с библиотеками NumPy, Matplotlib и PyGame для визуализации движения агентов, динамики тепловой карты и обхода препятствий. Алгоритм корректирует скорость дронов, учитывая посещаемость зон и среднюю скорость роя, что позволяет равномерно распределять агентов, избегать «пробок» и сокращать время поиска целей. Тестирование подтвердило эффективность: дроны покрывают всю территорию, не закливаясь в локальных минимумах, и быстро возвращаются на базу после выполнения задачи. Перспективы работы включают интеграцию машинного обучения для адаптации алгоритмов в реальном времени и внедрение компьютерного зрения для распознавания объектов. Модель служит основой для тестирования других гибридных решений в управлении роевым интеллектом.*

*Ключевые слова:* беспилотные системы, роевой интеллект, гибридные алгоритмы, тепловая карта, случайное блуждание, синхронизация данных, имитационное моделирование, Python.

Современные технологии развиваются с невероятной скоростью, и одним из наиболее значимых достижений последних десятилетий стало появление беспилотных систем управления. Эти системы представляют собой совокупность программно-аппаратных решений, позволяющих управлять различными объектами – от дронов и наземных роботов до сложных промышленных механизмов без непосредственного участия человека. Они обеспечивают высокую автономность, точность и безопасность, снижая вероятность ошибок, связанных с человеческим фактором, а также способны повышать оперативность и экономическую эффективность выполнения задач.

Беспилотная система – это любое устройство или программно-аппаратный комплекс, способный функционировать без непосредственного управления человеком. К беспилотникам относятся дроны, наземные и подводные роботы, а также автоматизированные транспортные средства.

Современные беспилотные аппараты кардинально трансформируют подходы к решению задач в различных областях, поскольку их способность работать в зонах, недоступных для человека, делает их незаменимыми инструментами. Однако одиночные дроны сталкиваются с фундаментальными

ограничениями – малым радиусом действия, уязвимостью к помехам и неспособностью эффективно исследовать большие территории, что стимулирует развитие комплексных беспилотных систем.

Беспилотные системы находят применение в самых разных сферах: в авиации беспилотные летательные аппараты (БПЛА) используются для мониторинга территорий, разведки, логистики и даже доставки грузов; в военной сфере дроны и наземные роботы применяются для разведывательных миссий, ведения боевых действий, дистанционного разминирования местности и объектов, патрулирования и охраны объектов, подвоза боеприпасов, эвакуации раненых; в гражданской инфраструктуре автоматизированные системы обеспечивают контроль трафика, мониторинг состояния мостов, зданий и инженерных сооружений и коммуникаций; в сельском хозяйстве беспилотники помогают в мониторинге посевов, внесении удобрений, распылении пестицидов и учёте урожайности; для поисково-спасательных операций автономные дроны и роботы способны обследовать труднодоступные или опасные районы и искать пострадавших после стихийных бедствий, а в космических исследованиях позволяют дистанционно изучать поверхность планет, астероидов и других космических объектов.

Главные качества беспилотных аппаратов:

- автономность (способность функционировать без постоянного контроля оператора);
- адаптивность (возможность изменять своё поведение в зависимости от окружающей среды и задач);
- интерактивность (способность взаимодействовать с другими устройствами, операторами и внешними системами);
- высокая точность (способность к точному выполнению задач, например, при навигации, мониторинге местности или доставке грузов).

В результате анализа масштабов и сложности задач, требующих участия беспилотных аппаратов, были выявлены недостатки одиночных дронов:

- ограниченная зона покрытия (одиночный дрон тратит большое количество времени на перемещение между целями (объектами поиска), особенно в условиях пересеченной местности);
- риск потери данных (централизованные системы управления уязвимы к сбоям связи, например, в условиях городской застройки или горной местности, в условиях помех или радиоэлектронного подавления (РЭП));
- дублирование усилий (без кооперации дроны многократно исследуют одни и те же участки, снижая общую эффективность миссии);
- энергетическая неэффективность (неоптимальные маршруты сокращают время автономной работы).

Эти вызовы и развитие беспилотных технологий привели к разработке роевого интеллекта, вдохновленного поведением природных систем, таких как стаи птиц, рой пчёл, муравьиная колония и т. д. [1], [2].

Роевой интеллект – это концепция, основанная на принципах коллективного поведения автономных агентов (дронов, роботов, сенсоров), которые взаимодействуют между собой и принимают решения на основе распределенной системы управления [1].

Роевые алгоритмы [3] позволяют группе агентов действовать как единый организм, демонстрируя эмерджентные свойства: самоорганизацию, адаптацию к динамической среде и коллективное принятие решений без централизованного контроля.

Основные особенности роевого интеллекта:

- децентрализованное управление – рой не зависит от единого центра, каждый элемент принимает решения на основе локальной информации;
- самоорганизация – рой адаптируется к изменениям в среде без необходимости внешнего вмешательства;
- масштабируемость – количество участников в рое может варьироваться без потери эффективности;
- отказоустойчивость – выход из строя отдельных элементов не влияет на общую работу системы;
- коллективный разум – дроны обмениваются данными, оптимизируя свои действия и принимая решения совместно.

**Методы управления роем.** Управление децентрализованным роем беспилотных аппаратов может осуществляться с помощью различных алгоритмов, которые имеют свои преимущества и недостатки [4], некоторые из них представлены в табл. 1.

**Таблица 1**

**Алгоритмы управления роем дронов**

| <i>Алгоритм</i>     | <i>Преимущества</i>                     | <i>Недостатки</i>                                              |
|---------------------|-----------------------------------------|----------------------------------------------------------------|
| Роевая оптимизация  | Реализует глобальный поиск              | Склонен к застреванию в локальных минимумах                    |
| Муравьиный алгоритм | Находит оптимальные маршруты            | Требуется точной настройки испарения феромона                  |
| Пчелиный алгоритм   | Подходит для распределенного поиска     | Требуется балансировки числа агентов                           |
| Модель стаи птиц    | Простая организация агентов             | Склонен к отклонениям в поисках маршрутов                      |
| Иммунный алгоритм   | Адаптивность к изменениям среды         | Требуется дополнительного алгоритма для поддержки движения [6] |
| Эвристический рой   | Гибкость в выборе стратегий             | Ресурсоемкость                                                 |
| Динамический лидер  | Позволяет координировать рой            | Зависимость от механизма смены лидера                          |
| Потенциальные поля  | Хорошо избегает препятствия             | Склонен к застреванию в локальных минимумах                    |
| Тепловая карта      | Обеспечивает равномерное покрытие карты | Склонен к застреванию в локальных минимумах                    |

## Окончание табл. 1

| Алгоритм            | Преимущества                          | Недостатки                                                    |
|---------------------|---------------------------------------|---------------------------------------------------------------|
| Случайное блуждание | Помогает избегать локальных минимумов | Ресурсоемкость и нет гарантии нахождения глобального оптимума |

Таким образом, многие алгоритмы управления роем дронов при использовании имеют критические недостатки, такие как:

- застревание в локальных минимумах (значения целевой функции минимально в небольшой малопосещаемой окрестности, но не во всей исследуемой области, поэтому рой дронов будет сходиться в этом локальном минимуме без возможности его покинуть);

- ресурсоемкость (потребление большого количества ресурсов, таких как время вычислений, энергия, память, что критично для роя дронов);

- зависимость от механизмов работы алгоритма (подбор параметров для некоторых алгоритмов имеет высокую степень важности, а изменения плотности роя, максимальной скорости, радиуса обзора и радиуса связи дронов могут иметь критическое влияние на эффективность работы системы).

Для решения данных проблем используются гибридные алгоритмы, которые представлены в табл. 2, они комбинируют разные одиночные алгоритмы, объединяют их сильные стороны и минимизируют недостатки друг друга.

Таблица 2

## Гибридные алгоритмы управления роем дронов

| Комбинация алгоритмов                                 | Описание объединения                                                                           | Преимущества                                                           | Недостатки                                                                        |
|-------------------------------------------------------|------------------------------------------------------------------------------------------------|------------------------------------------------------------------------|-----------------------------------------------------------------------------------|
| 1                                                     | 2                                                                                              | 3                                                                      | 4                                                                                 |
| Муравьиный алгоритм + роевой оптимизационный алгоритм | Муравьиный алгоритм отвечает за поиск маршрутов, а рой корректирует движение агентов глобально | Быстрое нахождение кратчайших путей; адаптивность к изменяющейся среде | Требуется балансировка влияния обоих алгоритмов; высокая вычислительная сложность |
| Алгоритм тепловой карты + муравьиный алгоритм         | Агенты избегают часто посещаемых зон, но при этом строят маршруты через муравьиный алгоритм    | Хорошее покрытие карты; эффективное исследование территории            | Возможность застревания в "холодных" тупиках                                      |
| Алгоритм ведущего и ведомых + муравьиный алгоритм     | Ведущие строят маршруты, ведомые адаптируются                                                  | Быстрая адаптация к изменениям; хорошая маршрутизация                  | Возможность перегрузки лидеров                                                    |

## Окончание табл. 2

| 1                                                                                 | 2                                                                                                                                                | 3                                                        | 4                                  |
|-----------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------|----------------------------------------------------------|------------------------------------|
| Пчелиный алгоритм + алгоритм тепловой карты                                       | Пчелиный алгоритм распределяет агентов, тепловая карта направляет их в малоизученные зоны                                                        | Равномерное покрытие; баланс разведки и оптимизации      | Требует точного подбора параметров |
| Искусственная иммунная система + эвристический рой                                | Обучение агентов избеганию препятствий + гибкие решения                                                                                          | Самообучаемость; хорошо для сложных карт                 | Высокая вычислительная нагрузка    |
| Метод потенциальных полей + муравьиный алгоритм                                   | Потенциальные поля помогают избегать препятствий, а муравьи ищут оптимальные пути                                                                | Обход препятствий + эффективный маршрут                  | Возможность застревания            |
| Муравьиный алгоритм + роевой оптимизационный алгоритм + метод потенциальных полей | Муравьиный алгоритм строит маршруты, роевой оптимизационный алгоритм корректирует движение, а потенциальные поля помогают избегать препятствий   | Быстрая маршрутизация; эффективное избегание препятствий | Высокая вычислительная сложность   |
| Пчелиный алгоритм + алгоритм тепловой карты + искусственная иммунная система      | Пчёлы распределяют агентов, тепловая карта направляет их в нужные зоны, а искусственная иммунная система помогает избегать препятствий           | Хорошее покрытие карты; самообучение                     | Высокая сложность калибровки       |
| Алгоритм ведущего и ведомых + муравьиный алгоритм + алгоритм тепловой карты       | Лидер строит маршрут с муравьиный алгоритм, ведомые корректируют движение, избегая перегруженных зон                                             | Гибкость стратегии; хорошее покрытие карты               | Требует точного управления лидером |
| Муравьиный алгоритм + пчелиный алгоритм + роевой оптимизационный алгоритм         | Муравьиный алгоритм строит маршруты, пчёлы распределяют разведку, а роевой оптимизационный алгоритм корректирует движение группы                 | Оптимальное покрытие + эффективная маршрутизация         | Высокие вычислительные затраты     |
| Метод потенциальных полей + искусственная иммунная система + эвристический рой    | Потенциальные поля помогают обходить препятствия, искусственная иммунная система обучает агентов адаптации, эвристический рой выбирает стратегию | Самообучение + адаптация к динамике                      | Высокая сложность калибровки       |

Таким образом, комбинации различных алгоритмов помогают решать локальные проблемы некоторых одиночных алгоритмов, но приобретают свои недостатки, приемлемые для построения эффективной модели.

Какими качествами будет обладать готовое изделие, можно оценить по результатам проведения полигонных испытаний [5].

Полигонные испытания – это методы контроля, реализуемые на испытательных полигонах, чаще всего полигонный контроль применяется для испытаний различной техники [6], [7].

При этом в условиях ограничений временных и финансовых ресурсов натурные испытания можно заменить или комбинировать с имитационным моделированием [8].

На межвидовом испытательном полигоне имеется комплексная испытательная моделирующая установка (КИМУ) [9], которая позволяет моделировать одновременное движение нескольких сотен объектов в реальном масштабе времени. Однако задача оперативного управления траекториями этих объектов не предусмотрена.

Для разработки имитационной модели поведения роя дронов и визуализации работы алгоритмов был выбран язык программирования Python [10], [11], который отвечает всем требованиям для решения задач распределенных вычислений и визуализации. Для построения подобной модели могут быть использованы следующие библиотеки языка программирования Python.

1. NumPy – это фундаментальная библиотека для научных вычислений в Python. NumPy предоставляет эффективные многомерные массивы и широкий набор математических функций для их обработки. Благодаря своей производительности и удобству работы с данными она является основой для многих других библиотек и помогает обрабатывать числовые данные, необходимые для моделирования поведения агентов.

2. SciPy, построенная на базе NumPy. SciPy предлагает дополнительные алгоритмы и функции для оптимизации, интегрирования, интерполяции, статистики и решения систем линейных уравнений. Эта библиотека помогает проводить сложные вычислительные задачи и моделирование, что важно при создании симуляций распределённых систем.

3. Pandas специализируется на обработке и анализе данных. Она предоставляет удобные структуры данных, такие как DataFrame и Series, для хранения, фильтрации и агрегации информации. Это особенно полезно для управления данными, собранными в ходе работы роевого интеллекта, и для последующего анализа результатов.

4. Matplotlib – это библиотека для создания графиков и визуализации данных. С её помощью можно наглядно демонстрировать динамику системы, отслеживать изменения в поведении агентов и визуализировать результаты экспериментов. Это важный инструмент для анализа работы распределённых систем.

5. OpenCV – это мощная библиотека для компьютерного зрения. Она предоставляет инструменты для обработки изображений, распознавания объектов, видеоанализа и даже отслеживания движущихся целей. В контексте роевого интеллекта OpenCV может использоваться для визуального контроля за поведением агентов или для обработки входных данных с камер.

6. Scikit-learn – это библиотека для машинного обучения, содержащая широкий спектр алгоритмов для классификации, регрессии, кластеризации и снижения размерности. Scikit-learn позволяет применять алгоритмы для анализа и прогнозирования поведения агентов, а также для разработки систем, способных к самообучению.

7. TensorFlow – это фреймворк для глубокого обучения и численных вычислений. Он поддерживает разработку и обучение нейронных сетей, что может быть использовано для реализации адаптивных алгоритмов в системах роевого интеллекта, где агенты обучаются на основе больших массивов данных.

8. PyTorch – это аналог TensorFlow. PyTorch известен своей гибкостью и динамичным вычислительным графом, что упрощает экспериментирование и разработку новых алгоритмов. Он активно используется для разработки моделей машинного обучения, способных адаптироваться в условиях распределённых систем.

9. NetworkX – это библиотека, предназначенная для создания, анализа и визуализации графов и сетей. NetworkX отлично подходит для моделирования коммуникаций между агентами, анализа топологии сети роевого интеллекта и исследования влияния взаимосвязей на коллективное поведение.

10. Dask позволяет масштабировать вычисления, распределяя задачи между ядрами процессора или узлами кластера. Он интегрируется с NumPy, Pandas и другими библиотеками, что делает его удобным для обработки больших объёмов данных и симуляций в распределённых системах.

11. PyGame – это набор модулей (библиотек) языка программирования Python, предназначенный для написания мультимедиа-приложений, позволяющий построить визуализацию работы различных алгоритмов.

Таким образом, язык программирования Python в комбинации с различными библиотеками является мощным инструментом для решения различных инженерных, научных и других задач.

Для реализации управления движением роем дронов предлагается использовать гибридный алгоритм случайного блуждания с коррекцией на заполнение тепловой карты.

Тепловая карта (англ. heatmap) [12] – метод графического представления данных, где индивидуальные значения в таблице отображаются при помощи цвета. Удобно применять в Кластерном анализе (англ. cluster analysis) [13] – многомерная статистическая процедура, выполняющая сбор данных, содержащих информацию о выборке объектов, и затем упорядочивающая объекты в сравнительно однородные группы.

Для движения дронов используется случайное блуждание в качестве основного источника скорости агентов:

$$\begin{aligned}v_x &= v_x + (rnd - 0.5) \times stochastic\_factor, \\v_y &= v_y + (rnd - 0.5) \times stochastic\_factor,\end{aligned}$$

где  $rnd$  – случайное число от 0 до 1;  $stochastic\_factor$  – это случайное число, которое определяет интенсивность случайных изменений.

После стохастического изменения скорости агентов происходит поиск наиболее холодного участка карты, который корректирует движения агентов к наименее посещенным зонам:

$$p_i = \frac{1}{1+T_i},$$

где  $p_i$  – вероятность движения агента в область  $i$ ;  $T_i$  – количество посещений зоны.

Далее дроны, находящиеся в пределах радиуса связи, обмениваются информацией о своей скорости, что влияет на их движение и формирует среднюю скорость роя:

$$\begin{aligned}v_{avgx} &= \frac{\sum_n(vx_i \cdot w_i)}{\sum w_i}, \\v_{avgy} &= \frac{\sum_n(vy_i \cdot w_i)}{\sum w_i},\end{aligned}$$

где  $w_i = \max\left(0, 1 - \frac{dist}{comm\_radius}\right)$  – весовой коэффициент соседа, который уменьшается с увеличением расстояния;  $dist$  – расстояние между агентами;  $comm\_radius$  – радиус связи между агентами;  $vx_i, vy_i$  – компоненты скорости соседа.

Каждый дрон самостоятельно производит корректировку по средней скорости всех агентов роя:

$$\begin{aligned}v_x &= v_x + s \times v_{avgx} - v_x \\v_y &= v_y + s \times v_{avgy} - v_y,\end{aligned}$$

где  $s$  – сила выравнивания.

Для синхронизации данных о целях предлагается использовать блокчейн [14] – это выстроенная по определённым правилам непрерывная последовательная цепочка блоков (связный список), содержащая какую-либо информацию, данный инновационный подход имеет преимущество над традиционными методами в виде децентрализованной работы, защиты от подделки данных и отказоустойчивости, но имеет недостаток в виде высокой ресурсоёмкости. При реализации такой синхронизации для каждого дрона был использован следующий подход:

- получить данные о целях от ближайших дронов-соседей;
- проверить состояние блокчейна на обновления;
- обновить состояние целей в блокчейне.

Для синхронизации тепловых карт, пройденного маршрута и других данных предлагается механизм CRDT (Conflict-Free Replicated Data Types), который реализует различные структуры данных (счетчики, множества,

тексты). Он гарантирует согласованность действий без явного разрешения конфликтов. Из недостатков выделяют высокое потребление памяти, но это решается путём удаления дубликатов при каждом обновлении данных.

**Результаты моделирования.** Обеспечивается визуализация имитационной модели роевого интеллекта, использующего синхронизацию данных [15], [16], с помощью объединения механизмов блокчейна и CRDT, которые реализуют необходимый и достаточный обмен данными между дронами, а также комбинированный алгоритм случайного блуждания дронов с коррекцией на тепловую карту.

Данная комбинация алгоритма и синхронизации минимизирует недостатки путём нивелирования основной проблемы тепловых карт – склонности к заикливаниям в локальных минимумах с помощью хранения и постоянной синхронизации менее посещённых мест на карте.

Рассмотрим концепцию разработанной модели.

Генерируется карта с визуализацией препятствий, целей (объектов поиска) со случайным местоположением их на карте, агентов (дронов) с радиусами обзора (желтые окружности) и связи (белые окружности), базой для них (синий квадрат в левом верхнем углу). Начальный этап работы имитационной модели отображён на рис. 1.

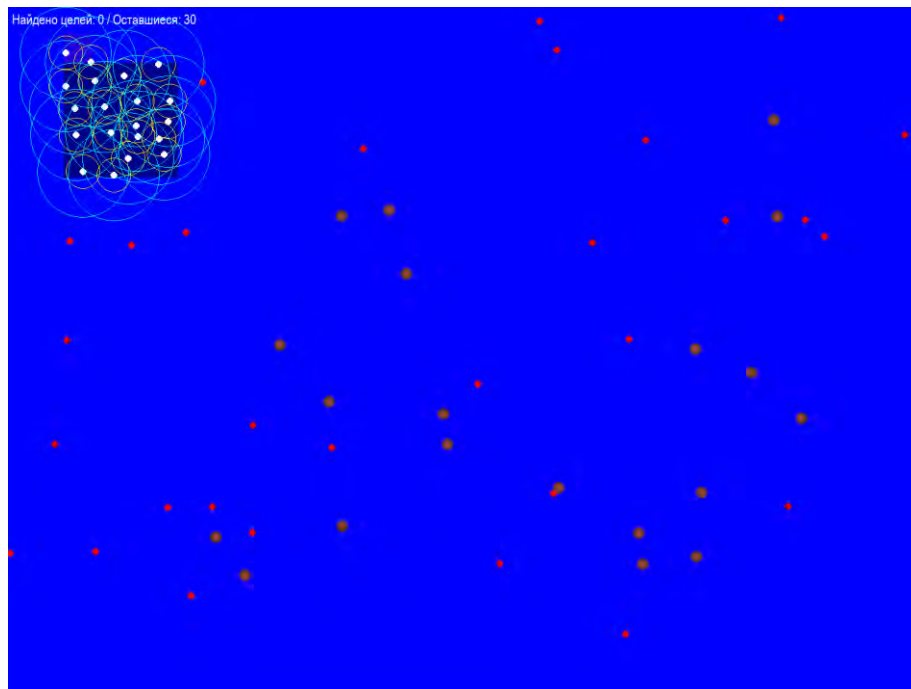


**Рис. 1. Начальный этап работы имитационной модели**

При запуске симуляции алгоритм начинает работу и модель визуализирует посещаемость зон на тепловой карте, где ярко-красный цвет соответствует наиболее частым посещениям, а синий соответствует непосещённым зонам.

Основное движение дронов для прохода ещё не посещённых зон тепловой карты представлено на рис. 2.

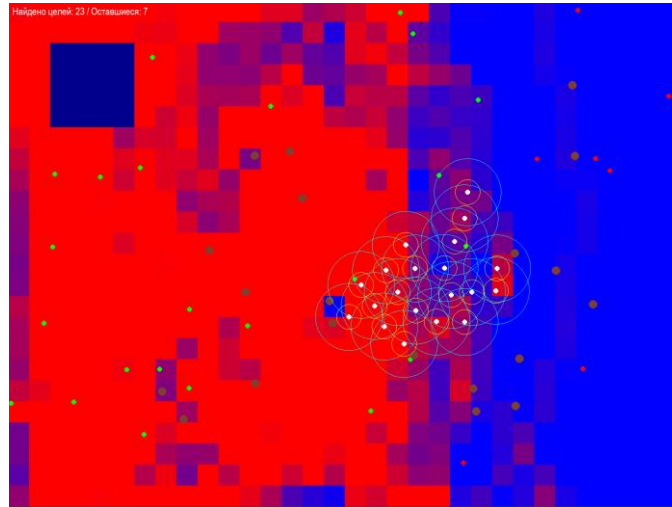


**Рис. 2. Визуализация зон посещений на тепловой карте**

Для обхода препятствий на пути агенты (дроны) изменяют своё направление:

$$\begin{aligned} dx &= x_{agent} - x_{obs}, \\ dy &= y_{agent} - y_{obs}, \\ angle &= atan2(dy, dx), \\ v_x &= v_x + \cos(angle) \times 0.1, \\ v_y &= v_y + \sin(angle) \times 0.1, \end{aligned}$$

где  $dx$ ,  $dy$  – вектор отклонения от препятствий;  $atan2(dy, dx)$  – функция для вычисления угла отклонения.

После нахождения всех условных целей на экран выводится время работы алгоритма, и дроны при получении информации о нахождении всех целей возвращаются на базу, что продемонстрировано на рис. 3.

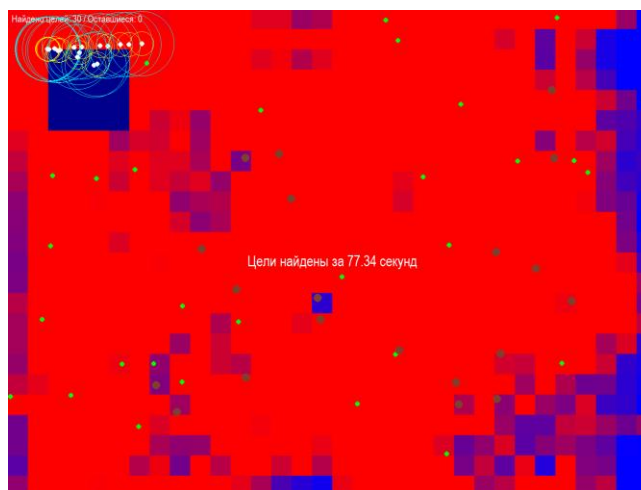


**Рис. 3. Результат работы алгоритма**

**Выводы.** В ходе реализации множества симуляций в имитационной модели гибридного алгоритма случайного блуждания агентов роя с коррекцией на заполнение тепловой карты было выявлено, что данное сочетание синхронизаций и алгоритмов позволяет проходить все зоны и составлять карту местности, организовывать поиск необходимых целей без попадания в локальные минимумы и за приемлемое время.

Предложенная имитационная модель может быть использована как основа для тестирования различных сочетаний синхронизации и алгоритмов управления роевым интеллектом. В дальнейшем планируется модернизация программного обеспечения для возможности выявления самых оптимальных алгоритмов синхронизации и управления роем дронов для использования компьютерного зрения, моделирования процесса поиска и картографирования на реальных данных, для расширения возможностей моделей с помощью внедрения машинного обучения и для анализа данных, получаемых от разных алгоритмов синхронизации и управления.

### Список литературы

1. Каляев И. А., Гайдук А. Р., Капустян С. Г. Роевой интеллект в распределенных системах управления. М.: Физматлит, 2016. 288 с.
2. Лебедев Б. К., Лебедев К. Б. Алгоритмы коллективного поведения в задачах навигации мобильных роботов // Информационные технологии. 2018. Т. 24. № 5. С. 312 – 319.
3. Храмов П. В., Семенов В. В. Многоагентные системы и роевой интеллект: теория и приложения. СПб.: Политехника, 2020. 415 с.
4. Матвеев А. С., Сахаров А. В. Децентрализованное управление группой БПЛА на основе роевого интеллекта // Вестник МГТУ им. Н.Э. Баумана. Сер.: Приборостроение. 2021. № 2(143). С. 78 – 92.
5. Бориско С. Н., Горемыкин С.А. Полигонные испытания как этап жизненного цикла сложной информационной системы // Известия Волгоградского государственного технического университета. 2016. № 11. Вып. 190. С. 27 – 31.
6. Бориско С. Н., Мустафаев Н. Г., Соколов С. П. Комплексный метод оценивания характеристик сложных систем: монография. Глава 9. Петрозаводск: МЦНП «Новая наука», 2021. 191 с.
7. Повышение точности оценивания параметров траектории высокоманевренных воздушных объектов на основе разложения по ортогональным функциям / В. И. Лобейко, С. Н. Бориско, С. В. Веселов, С. А. Князев, Н. Г. Мустафаев // Вестник воздушно-космической обороны. 2020. № 1. Вып. 25. С. 25 – 28.
8. Синтез методов испытаний сложных технических систем. Вопросы современной науки: монография / В. И. Лобейко, Н. Мустафаев, А. В. Старусев, Л. А. Михолап. М.: ООО "Интернаука". 2019. 120 с.

9. Об условиях оптимальной работы операторов в моделирующих установках специального назначения / В. И. Лобейко, А. В. Старусев, Л. А. Михолап, С. П. Литвинов // Сборник статей конференции "Теория и практика модернизации научной деятельности в условиях цифровизации". Стерлитамак: Международный центр инновационных исследований "OMEGA SCIENCE", 2020. С. 45 – 50.

10. URL: <https://github.com/pygame/pygame> (дата обращения 07.03.2025).

11. Федоренко Р. В. Применение генетических алгоритмов и роевого интеллекта в задачах оптимизации // Системный анализ и прикладная информатика. 2020. № 4. С. 22 – 30.

*Бородин Иван Владиславович, инженер-испытатель, [deuscontramundum@yandex.ru](mailto:deuscontramundum@yandex.ru), Россия, Знаменск, 4-й Государственный центральный межвидовой полигон Министерства обороны Российской Федерации,*

*Коротков Антон Анатольевич, инженер-испытатель, [korotkov.dev@yandex.ru](mailto:korotkov.dev@yandex.ru), Россия, Знаменск, 4-й Государственный центральный межвидовой полигон Министерства обороны Российской Федерации,*

*Бориско Сергей Николаевич, канд. техн. наук, доцент, зав. кафедрой, [boris62ko@mail.ru](mailto:boris62ko@mail.ru), Россия, Знаменск, Астраханский государственный университет имени В. Н. Татищева,*

*Тимошкин Александр Александрович, канд. техн. наук, нач. отдела, [timoshkin.a.a@yandex.ru](mailto:timoshkin.a.a@yandex.ru), Россия, Знаменск, 4-й Государственный центральный межвидовой полигон Министерства обороны Российской Федерации*

#### *SIMULATION OF A DECENTRALIZED DRONE SWARM TO OPTIMIZE SEARCH OPERATIONS IN AN UNEXPLORED AREA*

*S. N. Borisko, I. V. Borodin, A. A. Korotkov, A. A. Timoshkin*

*The article presents a hybrid algorithm for controlling a swarm of drones, combining a random walk with a correction to a heat map. The goal is to solve the problems of classical methods: getting stuck in local minima, duplicating routes, and uneven coverage of the search area. Blockchain (decentralization, anti-counterfeiting) and CRDT (conflict-free synchronization) are used to synchronize data between drones. The simulation is performed in Python with NumPy, Matplotlib, and PyGame libraries for visualizing agent movement, heat map dynamics, and obstacle avoidance. The algorithm adjusts the speed of the drops, taking into account the attendance of the zones and the average swarm speed. This allows you to evenly distribute agents, avoid "traffic jams" and reduce the time needed to find targets. Testing has confirmed its effectiveness: drones cover the entire territory without getting stuck in local minima, and quickly return to base after completing the task. The work prospects include the integration of machine learning to adapt algorithms in real time and the introduction of computer vision for object recognition. The model serves as the basis for testing other hybrid swarm intelligence management solutions.*

*Key words: unmanned systems, swarm intelligence, hybrid algorithms, heat map, random walk, data synchronization, simulation, Python.*

*Borodin Ivan Vladislavovich, test engineer, [deuscontramundum@yandex.ru](mailto:deuscontramundum@yandex.ru), Russia, Znamensk, 4th State Central Interspecific Training Ground of the Ministry of Defense of the Russian Federation,*

*Korotkov Anton Anatolyevich, test engineer, [korotkov.dev@yandex.ru](mailto:korotkov.dev@yandex.ru), Russia, Znamensk, 4th State Central Interspecific Training Ground of the Ministry of Defense of the Russian Federation,*

*Borisko Sergey Nikolaevich, candidate of technical sciences, docent, head of the chair, [boris62ko@mail.ru](mailto:boris62ko@mail.ru), Russia, Znamensk, Astrakhan State University named after V. N. Tatishchev,*

*Timoshkin Alexander Alexandrovich, candidate of technical sciences, head of the department, [timoshkin.a.a@yandex.ru](mailto:timoshkin.a.a@yandex.ru), Russia, Znamensk, 4th State Central Interspecific Training Ground of the Ministry of Defense of the Russian Federation*