

И.В. Бычков, А. В. Давыдов, М. Ю. Кензин, Н. В. Нагул
Институт динамики систем и теории управления СО РАН им. В.М. Матросова
Россия, 664033, г. Иркутск, ул. Лермонтова, 134

ИЕРАРХИЧЕСКОЕ ПЛАНИРОВАНИЕ ДЕЙСТВИЙ РАЗНОРОДНОЙ ГРУППЫ АВТОНОМНЫХ МОБИЛЬНЫХ РОБОТОВ

I. V. Bychkov, A. V. Davydov, M. Yu. Kenzin, N. V. Nagul
¹Matrosov Institute for System Dynamics and Control Theory
Russia, 664033, Irkutsk, Lermontov str., 134

HIERARCHICAL PLANNING OF ACTIONS OF A HETEROGENEOUS GROUP OF AUTONOMOUS MOBILE ROBOTS

I. V. Бичков, А. В. Давидов, М. Ю. Кензін, Н. В. Нагул
¹Інститут динаміки систем та теорії управління СО РАН ім. В.М. Матросова
Росія, 664033, м. Іркутськ, вул. Лермонтова, 134

ІЄРАРХІЧНЕ ПЛАНУВАННЯ ДІЙ РІЗНОРІДНОЇ ГРУПИ АВТОНОМНИХ МОБІЛЬНИХ РОБОТІВ

В статье представлены общие черты многоуровневого подхода к планированию миссии групп автономных мобильных роботов. Подход совмещает в себе верхнеуровневое планирование на основе эволюционных алгоритмов и логический уровень, опирающийся на формализацию логических дискретно-событийных систем в оригинальном исчислении позитивно-образованных формул. Областью применения разрабатываемого подхода являются системы управления техническими системами, прежде всего, робототехническими комплексами.

Ключевые слова: позитивно-образованная формула, автоматическое доказательство теорем, прuver, дискретно-событийная система, задача планирования рабочих смен, эволюционные алгоритмы, эвристики

The article presents the general features of a multi-level approach to mission planning for groups of autonomous mobile robots. The approach combines top-level planning based on evolutionary algorithms and a logical level based on the formalization of logical discrete-event systems in the original calculus of positively formed formulas. The area of application of the developed approach is control systems for technical systems, primarily robotic complexes.

Key words: positively-constructed formula, automatic theorem proving, prover, discrete event system, work shift scheduling problem, evolutionary algorithms, heuristics.

У статті наведено загальні риси багаторівневого підходу до планування місії груп автономних мобільних роботів. Підхід поєднує в собі верхньорівневе планування на основі еволюційних алгоритмів та логічний рівень, що спирається на формалізацію логічних дискретно-подійних систем в оригінальному обчисленні позитивно-освічених формул. Областю застосування підходу, що розробляється, є системи управління технічними системами, насамперед, робототехнічними комплексами.

Ключові слова: позитивно-утворена формула, автоматичний доказ теорем, прuver, дискретно-подієва система, задача планування робочих змін, еволюційні алгоритми, евристики

Введение

Технологический скачок последних десятилетий в области робототехники стимулировал новые тенденции в развитии автономных мобильных систем, крупнейшей из которых является объединение нескольких, зачастую разнородных транспортных средств для коллективного выполнения общей задачи. Применение группы автономных мобильных роботов (АМР) требует решения ряда новых комплексных задач, так как качество работы группы в динамичной среде в большей степени зависит от эффективности схем внутригруппового взаимодействия, чем от надежности отдельных членов команды. В работе предлагается иерархический подход к организации системы динамического управления разнородной группой АМР, нацеленный на решение некоторых проблем и задач групповой координации.

В общем случае миссия группы АМР заключается в том, чтобы роботы выполнили некоторый набор заданий в пределах обозначенной области в соответствии с текущими эксплуатационными требованиями [1]. Эта проблема распределения задач и ресурсов в ее простейшей постановке может быть математически сформулирована в терминах задачи о назначениях, а в более общем случае – задачи маршрутизации транспорта. Как и любая модель реального мира, такая постановка с достаточной степенью детализации будет включать сложный набор взаимосвязанных ограничений различной природы, что делает практически невозможным ее решение в явном виде и за разумное время [2]. Это приводит к необходимости применения иерархического подхода, когда ряд ограничений выносится на более высокий уровень, чтобы обеспечить их априорное удовлетворение ценой некоторых потерь в общей эффективности конечного решения [3].

Обзор робототехнической литературы [4] показывает, что в качестве общегрупповых ограничений, работа с которыми требует особого подхода, могут быть выделены коммуникационные [5-7] и топливные (энергетические) [8], [9] ограничения. Несмотря на огромное количество различных подходов к решению каждого из этих двух ограничений, в литературе почти не встречается работ, в которой они использовались бы одновременно и применялись к функционально гетерогенной группе роботов. Подход, способный решать эту задачу, должен сочетать в себе ключевые особенности обеих областей, чтобы гибко и интеллектуально управлять событиями, связанными с энергетическими и коммуникационными потребностями группы. В этом смысле управление временем становится фундаментальным требованием для успешного выполнения сложной робототехнической миссии [10].

Для символической обработки данных, поступающих в систему или генерируемых ею, и формирования на их основе событий, вызывающих смену режимов функционирования группы АМР, предлагается использовать логический вывод в исчислении позитивно-образованных формул (ПОФ). Автоматическое доказательство теорем (АДТ) является активно развивающейся областью искусственного интеллекта, основанной на методах математической логики, что делает его наиболее формализованным направлением дедуктивного построения и автоматизированного вывода. Современной областью применения АДТ, среди прочего, является робототехника, где оно помогает в основном при планировании [11] и принятии решений [12]. Например, в [13] для планирования и управления роем в робототехнике используется язык PDDL, основанный на классическом АТР в стиле STRIPS.

Исчисление ПОФ является полным методом для АДТ с функциональными символами [14-16], и его основные приложения лежат в области управления динамическими и интеллектуальными системами, включая ориентацию телескопов [17],

управление лифтами, преследование движущихся целей [18], достижимость целевого множества [19] и др. Благодаря своим особенностям, исчисление ПОФ позволяет совмещать автоматический поиск логических выводов со специальными эвристиками, настраиваемыми для решаемой задачи. В задачах группового управления роботами с помощью метода опровержения ПОФ, например, могут исследоваться свойства безопасности и живучести в рамках автоматической проверки допустимости стратегии переключения режимов функционирования группы.

В статье будут предложен эволюционный подход к высокоуровневому планированию динамических миссий на основе событий и возможное применение АДТ в исчислении ПОФ на разных уровнях управления группой АМР при выполнении поставленных задач.

Схема групповой координации

Основная идея работы в контексте планирования групповых миссий – предложить универсальную, многофункциональную, гибкую и масштабируемую структуру системы управления, которая скорее будет опираться на ряд жестких ограничений и требований, свойственных большинству долгосрочных динамических миссий, чем на какие-либо конкретные задачи и типы действующих роботов.

Природа этих ограничений и формирует основную концепцию подхода: разбить миссию на последовательность рабочих периодов, на каждом из которых группа будет функционировать в неизменном составе, а все перестановки в составе будут осуществляться во время заранее запланированных сборов группы в конце каждого рабочего периода (рис. 1). Проведение таких сборов с требуемой периодичностью обеспечит регулярную коммуникацию внутри группы, а вставка дополнительных внеочередных сборов позволит своевременно отправлять нуждающихся роботов на подзарядку и принимать их обратно.



Рисунок 1 – Схема организации работы группы на основе рабочих периодов и групповых сборов

Применение такого подхода позволит разгрузить основной планировщик действий группы, который обычно отвечает за вычислительно трудоемкие задачи распределения заданий и составления маршрутов, передав некоторые из его наиболее жестких ограничений на решение задачи планирования на более высоком уровне управления.

В качестве еще одного важного уровня иерархической системы группового управления должен быть выделен обработчик событий, отвечающий за переключение режимов функционирования АМР на основе возникновения как запланированных, так и непрогнозируемых событий. Эффективная реализация такого логического обработчика событий позволит группе корректно реагировать на любые динамические изменения состояния как внешней среды, так и самих членов группы. Таким

образом, предлагаемый иерархический подход к организации групповой работы и соответствующая ей система группового управления могут быть представлены в виде следующей блок-схемы (рис. 2):

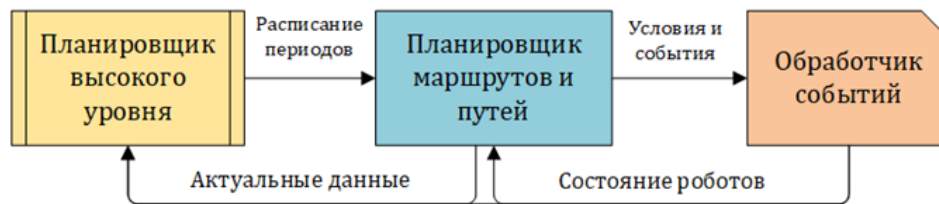


Рисунок 2 – Схема системы группового управления АМР в динамических условиях

Планировщик рабочих периодов

Задача планировщика на верхнем уровне – обеспечить эффективную ротацию группы на протяжении всей миссии в динамических условиях и с учетом требований к регулярности связи путем построения эффективного расписания рабочих периодов. Для построения такого расписания планировщик должен распределить во времени последовательность групповых сборов и поставить каждой из них в соответствие событие, связанное с подзарядкой АМР или проведением сеанса связи. При этом необходимо учитывать количество доступных систем подзарядки и их мощность.

Пусть для выполнения миссии длительностью T (горизонт планирования) имеется группа из n АМР, различающихся по крейсерской скорости v_i , емкости батарей \bar{b}_i и набору установленного оборудования (манипуляторы, датчики и др.) $\{u_i^{(j)}\} \in \{0,1\}, j = 1, \dots, l, i = 1, \dots, n$. Подзарядка батарей АМР производится со скоростью c на зарядной станции, оснащенной w зарядными доками и расположенной на расстоянии d от области группового сбора. Скорость зарядки c означает, что АМР заряжается в c раз быстрее, чем разряжается.

Обозначим за $Q_i = (t_{i1}, t_{i2})$ период подзарядки i -го робота как временной отрезок между групповым сбором, на которой АМР покидает группу для пополнения запаса энергии, и сбором, когда он возвращается обратно в группу. Аналогичным образом, интервал между двумя последовательными периодами подзарядки АМР будем называть его рабочей сменой. Ставится задача поиска наиболее эффективного сочетания допустимых рабочих смен для всей группы действующих роботов на период T :

$$S = \{Q_{11}, Q_{12}, \dots, Q_{21}, Q_{22}, \dots, Q_{n1}, Q_{n2}, \dots, Q_{nq}\}.$$

Каждый раз, когда какой-либо робот должен покинуть группу для подзарядки или вернуться в нее после, он инициирует проведение группового сбора. Проведение сбора, в свою очередь, означает переход на следующий рабочий период, где группа будет выполнять задачи в неизменном составе. Требование к периодичности связи накладывает дополнительное ограничение P на максимальную длительность каждого рабочего периода. Для удовлетворения этого ограничения без инициации избыточно частых перезарядок АМР, допускается проведение сборов только с целью коммуникационного обмена без изменения состава действующей группы (рис. 1).

Расписание рабочих смен группы S считается допустимым, если оно обеспечивает своевременную подзарядку всех роботов с учетом количества доступных док-станций:

$$\begin{aligned} \forall t \in [0, T] \{ |Q \in S: t \in Q| \} &\leq w, i = 1, \dots, n, \\ b_i(t) &> 0, t \in [0, T], i = 1, \dots, n, \\ b_i(t) &= b_i(t-1) + a, 0 \leq b_i(t) \leq \bar{b}_i, \end{aligned}$$

$$a = \begin{cases} c & t \in Q \\ -1 & t \notin Q \end{cases}$$

Здесь последние два уравнения определяют динамику изменения уровня заряда аккумуляторов $b_i(t)$ каждого робота во времени (рис. 3). В работе применяется абстрактная линейная модель энергопотребления, но в дальнейшем планируется переход к применению более продвинутых и реалистичных моделей [20].

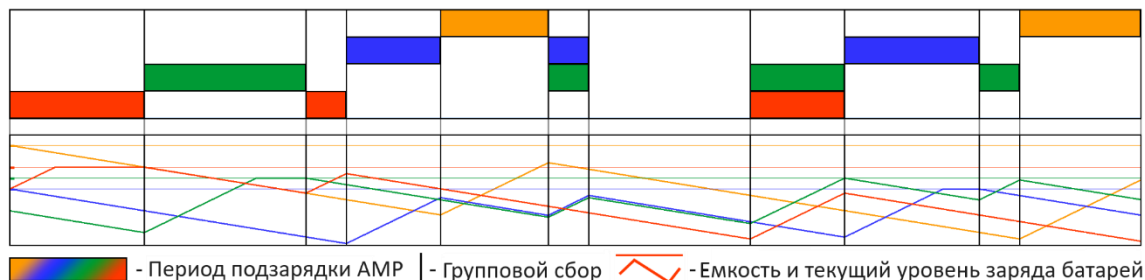


Рисунок 3 – Расписание периодов подзарядки группы из четырех разнородных АМР и соответствующий ему график прогноза энергетических ресурсов каждого робота

Эффективность расписания S определяется как совокупная эффективность работы всех подгрупп АМР, действующих на различных рабочих периодах. В отрыве от конкретной задачи, решаемой на среднем уровне, эффективность подгруппы может быть оценена через потенциальную работоспособность подгруппы, определяемую ее количественным и качественным составом. При условии, что максимальная эффективность достигается, когда в действующую группу входят все n роботов (ни один АМР не находится на подзарядке), задача заключается в минимизации временных и ресурсных потерь, вызванных как периодическими уходами АМР из группы на подзарядку, так и необходимостью регулярных групповых сборов, поскольку их проведение также отвлекает всех роботов от выполнения текущих задач.

Таким образом, групповое расписание должно отвечать следующему перечню критериев, где первый является критерием допустимости расписания, а остальные три – критериями его эффективности:

1. Своевременная подзарядка для всех роботов в группе;
2. Минимизация количество недоступных АМР на каждом рабочем периоде;
3. Запрет на одновременную подзарядку всех носителей каждого вида бортового оборудования (если их больше одного);
4. Минимальное количество необязательных (излишних) групповых сборов.

Рассматриваемая задача по своей сути является оригинальной вариацией известного класса задач комбинаторной оптимизации, объединяющего в себе множество родственных постановок, связанных с планированием расписания смен рабочего персонала (shift-, workforce-, employee-, personnel-, staff-scheduling) [21]. Суть задач этого класса состоит в том, чтобы выбрать набор сотрудников из имеющегося списка кандидатов и составить для каждого из них график смен таким образом, чтобы в пределах горизонта планирования покрыть потребности в рабочей силе с наименьшими затратами. При этом должны быть соблюдены требования к графику работы каждого сотрудника, такие как продолжительность смен, выходные дни и др. В этих задачах требования к закрытию всех потребностей обычно являются жесткими (обязательными к исполнению), а ограничения на формирование рабочей нагрузки сотрудников – мягкими, которые могут быть нарушены [22].

Сформулированная выше задача планирования может быть рассмотрена как обратная задача планирования рабочих смен, в которой состав сотрудников уже сформирован, ограничения на их рабочую нагрузку являются жесткими, а в качестве критерия выступает качество конечного покрытия потребностей (максимальная работоспособность действующей группы в каждый момент времени). Среди других особенностей предложенной модели – многопрофильные работники и повышенные требования к гибкости расписания, обусловленные динамическим характером задачи, что ведет к отказу от необходимости формирования решений в циклическом виде [23].

Алгоритм планирования рабочих периодов

Несмотря на большое разнообразие существующих подходов к задачам составления расписания смен в различных вариациях, эволюционные алгоритмы (ЭА) на протяжении уже многих лет остаются одним из наиболее популярных выборов. ЭА сочетают в себе высокую гибкость и масштабируемость с широкими возможностями по гибридизации оптимизационной схемы с другими проблемно-ориентированными процедурами и эвристиками [24]. Структура ЭА допускает раздельное управление исследованием и разработкой пространства поиска, чтобы фокусировать поиск на границе области допустимости, где для задач с плохой окрестностной структурой зачастую находятся наиболее качественные решения [25]. Естественный параллелизм ЭА позволяет ускорять формирование планов за счет распределения вычислений внутри группы взаимодействующих АМР, а популяция имеющихся кандидатов в решения выступает в качестве своеобразной структуры памяти, способствующей более быстрой корректировке планов в случае неожиданного изменения условий [26].

Исходя из вышесказанного, для решения поставленной задачи предлагается проблемно-ориентированная модификация эволюционных алгоритмов, обогащенная несколькими специализированными эвристиками и операторами улучшения.

В качестве хромосомы здесь выступает векторное представление решения в виде расписания S , а целевая штрафная функция имеет следующий вид:

$$F(S) = n \cdot V(S) + f(S) \rightarrow \min,$$

где $V(S)$ определяет степень нарушения ограничения на допустимость решения, а $f(S)$ оценивает отклонение работоспособности группы от максимально возможной. Функция $V(S)$ оценивает общую нехватку энергии АМР на реализацию расписания S по принципу «time-warp», разработанному для задач маршрутизации с временными окнами [27]. Цель использования этой функции – исследовать более широкое разнообразие структурно различных решений, разрешив добавлять в популяцию невыполнимые расписания.

Функция $f(S)$, в свою очередь, также является линейной сверткой из двух штрафных функций $f(S) = \mu \cdot f_E(S) + f_R(S)$, где первая функция оценивает потери коллективной работоспособности в связи с уходом АМР на перезарядку, а вторая – потери, связанные с проведением групповых сборов (μ – весовой коэффициент).

Большинство современных метаэвристических алгоритмов используют специально разработанные конструктивные эвристики для создания качественной начальной популяции решений [28]. Общее качество популяции определяется через три ее свойства: общее разнообразие, большой охват пространства поиска, наличие нескольких допустимых решений. Поэтому для первого шага ЭА были разработаны три различных конструктивных эвристики (рис. 4), одновременное использование которых позволяет алгоритму хорошо работать при различных комбинациях условий и ограничений задачи:

1. Первая эвристика (рис. 4а) генерирует непрерывную последовательность рабочих периодов, на каждом из которых группу покидает ровно один случайный АМР. Конечные расписания здесь зачастую являются недопустимыми, но всегда дают низкое значение штрафной функции $f_E(S)$;
2. Вторая эвристика (рис. 4б) гарантирует формирование допустимых решений ($V(S) = 0$) за счет следования двум правилам: ни одна рабочая смена АМР не длится дольше, чем емкость батареи робота; продолжительность периодов подзарядки АМР всегда превышает время его полной зарядки. Ограничение на количество зарядных доков в данном случае не учитывается.
3. Последняя эвристика (рис. 4с) направлена на генерацию разнообразных расписаний с малым количеством рабочих периодов (низким значением $f_R(S)$). Для этого сначала формируется последовательность максимально продолжительных рабочих периодов с учетом ограничения P , которые затем заполняются рабочими сменами АМР с учетом их энергетической потребности и количество док-станций w .

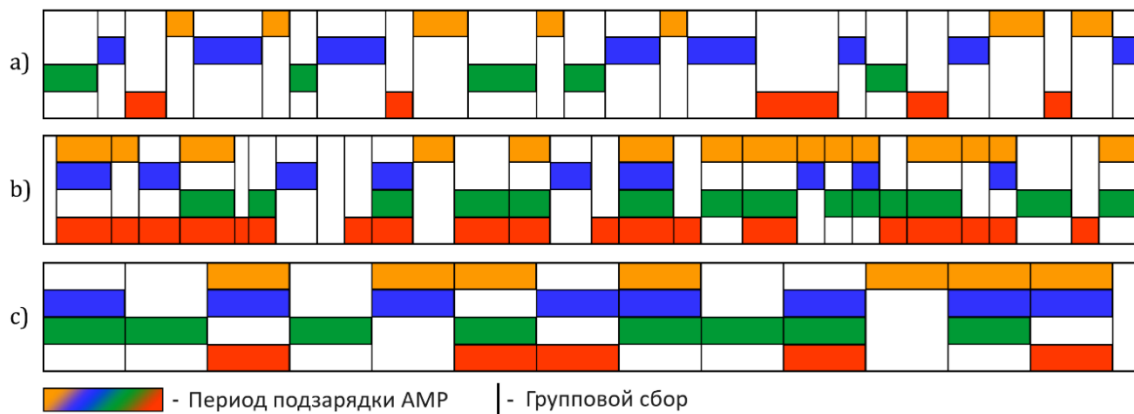


Рисунок 4 – Пример трех начальных решений различного типа для группы из 4 АМР

Для воспроизводства новых решений используется стандартная трехступенчатая схема отбора, скрещивания и мутации. Для отбора используется классическая схема дуэльного турнира, дающая больше шансов получить потомство разным по качеству решениям (особенно недопустимым).

Для получения решений-потомков применяется один из двух конкурирующих операторов скрещивания: классический одноточечный кроссовер либо однородное (uniform) скрещивание, для каждого АМР выбирающее расписание целиком от одного из решений-родителей. Затем применяется многорежимная мутация, состоящая из шести различных операторов, незначительно изменяющих текущее расписание.

В случае, если эволюционный процесс замедляется, полученные решения проходят через процедуру локального поиска по чередующимся окрестностям (VNS). Для этого были разработаны четыре оператора, нацеленных на устранение тех проблемных ситуаций, которые трудно исправить с помощью скрещивания и мутации.

Конечные решения-потомки подвергаются процедуре по восстановлению допустимости путем вставки пустых (без изменения состава группы) групповых сборов в середину тех рабочих периодов, длительностью которых превышает P .

Для формирования нового поколения решений применяется правило элитизма и модель островов (параллельных популяций). Процедура устранения решений-клонов используется дополнительно для поддержания разнообразия популяции.

Вычислительные эксперименты

Разработанная модель задачи планирования и алгоритм для ее решения были программно реализованы на языке C++ в системе моделирования «AUV Multiobjective Mission Planner» для проведения серии вычислительных экспериментов. С этой целью были разработаны два набора тестовых примеров: первый набор «test-A» построен вручную таким образом, чтобы априори знать глобальный оптимум, а набор «test-B» состоит из случайно сгенерированных комплексных задач большой размерности.

Для построения набора «test-A» был использован стандартный подход поиска циклического решения. Согласно этому подходу, строится кратчайший оптимальный подцикл подзарядки для всех АМР, который затем повторяется для заполнения всего расписания [29]. Рисунок 5 наглядно иллюстрирует типовой пример из «test-A», где $n = 4$, $w = 1$, $c = n - 1 = 3$, все АМР функционально идентичны. Очевидное оптимальное решение здесь основано на цикле (рис. 5a). Несмотря на кажущуюся простоту этого решения и множество глобальных оптимумов (порядок подзарядки может меняться), его практически невозможно найти подходами, основанными на исследовании пространства поиска, в следствие плохой окрестностной структуры, когда одни хорошие решения не находятся в окрестности с других. Другими словами, такие циклические решения являются слишком жесткими, так как любое самое незначительное его изменение сразу приводит к потере свойства допустимости [23]. Это означает, что вероятность получения таких решений путем применения генетических операторов или процедур локального поиска к другим допустимым решениям крайне мала. При этом стоит учитывать, что такие циклические решения могут быть легко построены только для самых простых примеров, а задача их генерации в общем случае, особенно для больших разнородных групп агентов, является открытой задачей [22].

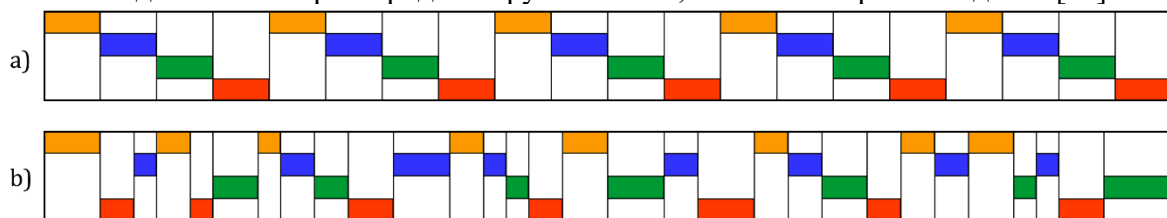


Рисунок 5 – Сравнение оптимального и приближенного решения тестового примера

Эволюционный алгоритм в таком случае в первую очередь начинает искать допустимые решения, а затем стремится повысить их эффективность (рис. 5b), по возможности уменьшая количество одновременных подзарядок нескольких АМР, а также минимизируя количество групповых сборов. Более короткие рабочие периоды в этом случае делают решение более гибким, позволяя осуществлять оптимизацию расписаний путем изменения длительности рабочих периодов и изменения их порядка.

На задачах из набора «test-B» подход тестировался на скорость и способность строить качественные решения в целом для задач большой размерности ($n \in [6,30]$, $l \in [4,10]$), требующих планирования до 50 рабочих периодов. Здесь алгоритм продемонстрировал хорошую масштабируемость и способность поддерживать общее количество смен при увеличении размера флота и усложнении ограничений. Он способен генерировать эффективные расписания разнородной группы из нескольких десятков АМР за несколько минут фоновых вычислений. При необходимости осуществления долгосрочного планирования для больших разнородных групп целесообразно ограничивать горизонт планирования некоторым среднесрочным окном, которое будет смещаться по мере выполнения миссии, чтобы избежать перегрузки вычислений.

Логический уровень системы управления

Уровень обработчика событий, отвечающий за переключение режимов функционирования АМР на основе возникающих событий, может быть реализован на основе одного из существующих формализмов логических дискретно-событийных систем (ДСС) [30]. Самыми известными из таких формализмов являются конечный автомат, а также близкая к нему по своей сути машина конечных состояний. Логическая ДСС в этом случае рассматривается как пятерка $\mathcal{G} = (Q, \Sigma, \delta, q_0, Q_m)$, где Q – множество состояний q ; Σ – множество событий; $\delta: \Sigma \times Q \rightarrow Q$ – функция переходов; $q_0 \in Q$ – начальное состояние; $Q_m \subset Q$ – множество выделенных (маркированных) состояний. Множество Q_m маркированных состояний используется для описания завершенных системой задач, например, миссии группы АМР или цепочки переключений режимов функционирования отдельного робота. Множество последовательностей событий образует формальный язык, описывающий функционирование системы на символическом уровне, а утверждения о свойствах этого языка могут быть получены в результате АДТ при соответствующей их формализации, например, как теоремы исчисления ПОФ. В частности, АДТ в исчислении ПОФ предложено использовать для анализа логических ДСС и синтеза супервизорного управления в рамках теории супервизорного управления [31].

Язык ПОФ представляет собой первопорядковый логический язык, состоящий из формул, построенных из атомарных с помощью связок \vee , $\&$, кванторов \forall и \exists , констант true и false. В общем виде ПОФ состоит из базы фактов и так называемых вопросов к базе, ответами на которые являются подстановки известных фактов вместо переменных вопроса. Консеквент вопроса, на который нашелся ответ, попадает в базу с соответствующей подстановкой. В результате построения вывода ПОФ как некоторой теоремы, в базе накапливаются факты, представляющие собой, например, план действий или, как в случае с ДСС, последовательность событий, возможных или произошедших в системе.

Для удобства чтения мы будем представлять ПОФ в виде дерева, узлами которого являются *типовые* кванторы. Относительно частей ПОФ будет использоваться соответствующая терминология: узел, корень, лист, ветвь. Например, ПОФ

$$\forall \{ \exists_{x_1} A_1 \{ \forall_{y_1} B_1, \forall B_2 \{ \exists_{x_2} A_2, \exists A_3 \{ \forall_{y_2} B_3 \}, \exists A_4 \} \}, \exists A_5 \{ \forall_{y_3} B_4 \} \}$$

представляется в виде дерева следующим образом:

$$\forall \left\{ \begin{array}{l} \forall_{y_1} B_1 \\ \exists_{x_1} A_1 \left\{ \begin{array}{l} \forall B_2 \left\{ \begin{array}{l} \exists_{x_2} A_2 \\ \exists A_3 - \forall_{y_2} B_3 \\ \exists A_4 \end{array} \right. \\ \exists A_5 - \forall_{y_3} B_4 \end{array} \right. \end{array} \right.$$

В формуле $P = \forall \{ F_1, \dots, F_n \}$ формулы $F_i = \exists_{x_i} B_i \{ Q_{i1}, \dots, Q_{im} \}$, $i = \overline{1, n}$, называются базовыми подформулами P , B_i называются базами фактов, или просто базами, Q_{ij} называются подформулами-вопросами, а корни подформул-вопросов – вопросами к базам B_i , $i = \overline{1, n}$. Вопрос вида $\forall_{x_i} A$ (без каких-либо потомков) называется целевым вопросом. Подробное описание исчисления ПОФ можно найти в [15].

Логическая ДСС автоматного типа \mathcal{G} может быть реализована с помощью ПОФ $\mathcal{F}_{\mathcal{G}}$ на рис. 6. База $\mathcal{F}_{\mathcal{G}}$ представляет собой множество $B_{\mathcal{G}} = \{ L(\varepsilon, q_0), L^m(\varepsilon, q_0), \delta(q_1^i, \sigma^i, q_2^i), \delta^m(q_1^j, \sigma^j, q_2^j) \}$, содержащее атомы, описывающие переходы между состояниями \mathcal{G} , в том числе маркированные, и атомы, которые будут использоваться

для построения генерируемых языков, $i, j \in \{1, \dots, n\}$, n – конечное число. Предикат $L(s, S)$ обозначает « s – это текущая последовательность событий, приведшая систему в состояние S », а предикат $L_m(s, S)$ обозначает « s – это текущая последовательность событий, приведшая систему в состояние S , и s – маркированная строка». Первые аргументы этих атомов накапливают строки языков $L(\mathcal{G})$ и $L_m(\mathcal{G})$, сгенерированного и маркированного автоматом \mathcal{G} , соответственно.

В зависимости от выбранного на текущем шаге вопроса, использование правила вывода ω исчисления ПОФ влечет генерацию в базе нового факта $L(s, q)$ или $L^m(s, q)$, при этом литеры s есть слово языка $L(\mathcal{G})$ или $L_m(\mathcal{G})$, соответственно.

$$\mathcal{F}_{\mathcal{G}} = \exists B_{\mathcal{G}} \left\{ \begin{array}{l} \forall \sigma, q, s, q' L(s, q), \delta(q, \sigma, q') - \exists L(s \cdot \sigma, q') \\ \forall \sigma, q, s, q' L^m(s, q), \delta^m(q, \sigma, q') - \exists L^m(s \cdot \sigma, q') \end{array} \right.$$

Рисунок 6 – Общая форма ПОФ-представления конечного автомата как логической ДСС

Особенностью предлагаемого подхода является возможность дополнительной обработки и контроля событий на основе данных об окружающей среде в режиме реального времени во время вывода. За это отвечают специальные логические правила, представленные в ПОФ в виде вопросов обработки событий. В результате ответа на них запускаются подвыводы, в которых поступающие данные служат параметрами, используемыми в расчетах или в других процессах принятия решений.

ПОФ в задаче перемещения блока

Рассмотрим задачу о перемещении объекта, которая может возникнуть в локальной области, куда была направлена часть группы АМР согласно расписанию верхнего уровня. Пусть в области, называемой далее сценой, находятся три робота, два блока и целевая область, в которую необходимо передвигать блоки (рис. 7). Будем полагать, что только два робота, действуя слаженно, могут сдвинуть блок, поэтому сначала робот должен найти другого робота для формирования пары, а затем толкать блок в целевую область. Для упрощения планирования сцена разделена на ячейки для отслеживания координат передвигаемых объектов. Рассматриваемая задача может быть описана набором конечных автоматов, каждый из которых фиксирует один из аспектов действий роботов. Один из этих автоматов представлен на рис. 7 и описывает смену основных режимов функционирования АМР.

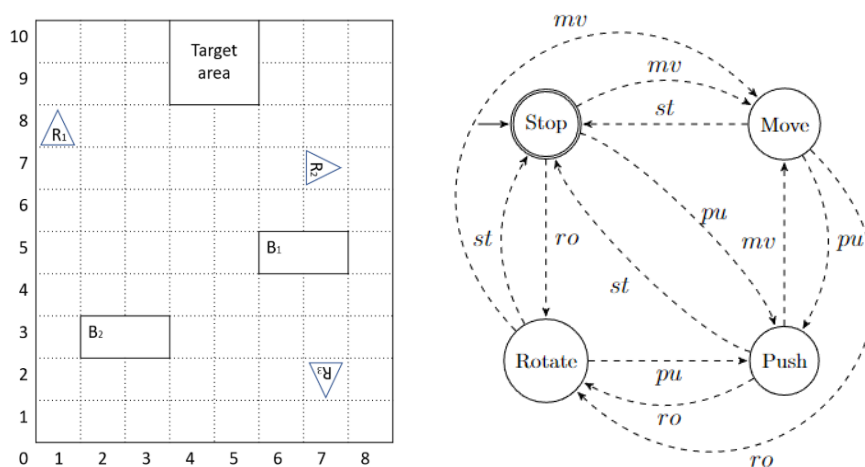


Рисунок 7 – Конечный автомат \mathcal{G}_{block} , описывающий смену режимов АМР в задаче перемещения блоков

В основанном на ПОФ подходе условия срабатывания переходов в \mathcal{G}_{block} формализуются в виде ПОФ, эквивалентных, в общем случае, нехорновским формулам логики предикатов первого порядка, что выходит за рамки формализации, доступной, например, в широко известном языке логического программирования Пролог. Такая ПОФ состоит из базы, которая содержит факты о начальном положении объектов на сцене, а также вопросов-правил, отвечающих за распределение целей для группы роботов, передвижения роботов к целям и перемещению блоков к целевой области. Для формализации будут использоваться следующие предикаты:

$R(x, y)$ – “ x – это робот, направленный в сторону $y \in \{N, E, S, W\}$ ”,

$B(x)$ – “ x – это блок, который нужно переместить”,

$W(x)$ – “ x – это ширина сцены”,

$H(x)$ – “ x – это высота сцены”,

$Pos(x, y, z)$ – “ x – это робот или блок, расположенный в координатах (y, z) ”,

$App(x, u, v)$ – “ x – робот, назначенный для выполнения задания по перемещению в координатах (u, v) ”,

$Free(x, y)$ – “координаты (x, y) свободны для перемещения туда робота”,

$Ready(x)$ – “ x готов переместить блок”,

$Move(x)$ – “ x перемещает блок”,

$BlockMoved(x, y)$ – “ x это блок с номером y который уже был перемещен”,

$\leq(x, y)$ – “вспомогательный вычисляемый предикат, который замещается логической константой *true* если x меньше или равен y , и константой *false* в обратном случае, мы будем использовать обычную инфиксную его форму”,

$=(x, y) \neq(x, y)$ – то же самое, что и выше, в отношении равенства и неравенства,

$dist(x, y, u, v)$ – функциональный символ, необходимый для оценки расстояния между ячейками с координатами (x, y) и (u, v) .

Предикат $Free()$ также будет считаться вычисляемым для удобства чтения и будет обозначаться символом $\#$. Символ $*$ рядом с предикатом указывает на то, что этот предикат будет удален из базы после успешного ответа, это подход к моделированию устаревания фактов с течением времени. База фактов содержит атомы, описывающие исходную сцену (рис. 7):

$$\exists R(r_1, N), R(r_2, E), R(r_3, S), B(b_1), B(b_2), Pos(r_1, 1, 8), Pos(r_2, 7, 7), Pos(r_3, 7, 2), \\ Pos(b_1, 6, 5), Pos(b_2, 2, 3), H(10), W(8)$$

В процессе поиска вывода в базу будут добавляться факты, отражающие последовательные изменения состояния системы. Например, в состоянии (режиме) “*Move*” при построении вывода в опровергнутых базах будут накапливаться факты, отражающие последовательные изменения состояния сцены, откуда можно извлечь минимальный план движения АМР. За остановку поиска вывода отвечает целевой вопрос-правило, сравнивающий координаты целевой области с текущими координатами блоков. Существенно, что в ходе вывода используются вспомогательные вопросы обработки событий. Первый вопрос в ПОФ $\mathcal{F}_G^\#$ на рис. 8 представляет собой общую форму вопроса, служащего для генерации событий, вызывающих переходы между состояниями.

$$\mathcal{F}_G^\# = \exists B_G \cup \{T^\#(\sigma)\} \left\{ \begin{array}{l} \forall \sigma T^\#(\sigma) - \exists E(\sigma) \\ \forall s, q, \sigma, q' L(s, q), E^*(\sigma), \delta(q, \sigma, q') - \exists L(s, \sigma, q') \end{array} \right.$$

Рисунок 8 – ПОФ, содержащая предикат $T^\#$ обработки события

Покажем ПОФ, используемые в предикате $T^\#$ для организации подвывода, проверяющего возникновение события “ ro ” для перехода АМР в режим вращения “ $Rotate$ ” (рис. 9).

$$\mathcal{F}_G^\# = \exists B_G \cup \{T^\#(p)\} \left\{ \begin{array}{l} \forall r, d T^\#(R(r, d)) - \exists E(ro) \\ \forall s, q, \sigma, q' L(s, q), E^*(ro), \delta(q, \sigma, q') - \exists L(s \cdot \sigma, q') \end{array} \right.$$

Рисунок 9 – ПОФ, содержащая предикат обработки события для перехода “ ro ”

Прежде всего, АМР необходимо повернуться, если расстояние между следующей позицией робота и текущей позицией цели меньше, чем начальная позиция робота:

1. $\forall r, x, y, u, v R^*(r, N), Pos(r, x, y), Free^\#(x, y + 1), App(r, u, v), dist(x, y, u, v) < dist(x, y + 1, u, v) \{ \exists R(r, E) \}$
2. $\forall r, x, y, u, v R^*(r, E), Pos(r, x, y), Free^\#(x + 1, y), App(r, u, v), dist(x, y, u, v) < dist(x + 1, y, u, v) \{ \exists R(r, S) \}$
3. $\forall r, x, y, u, v R^*(r, S), Pos(r, x, y), Free^\#(x, y - 1), App(r, u, v), dist(x, y, u, v) < dist(x, y - 1, u, v) \{ \exists R(r, W) \}$
4. $\forall r, x, y, u, v R^*(r, W), Pos(r, x, y), Free^\#(x - 1, y), App(r, u, v), dist(x, y, u, v) < dist(x - 1, y, u, v) \{ \exists R(r, N) \}$

При этом специальная группа правил задает эвристику для сокращения пространства поиска вывода при проверке расстояния между текущим положением робота и назначенной целью:

- e1. $\forall r, x, y, u, v R(r, N), Pos^*(r, x, y), Free^\#(x, y + 1), App(r, u, v), dist(x, y, u, v) \geq dist(x, y + 1, u, v) \{ \exists Pos(r, x, y + 1) \}$
- e2. $\forall r, x, y, u, v R(r, E), Pos^*(r, x, y), Free^\#(x + 1, y), App(r, u, v), dist(x, y, u, v) \geq dist(x + 1, y, u, v) \{ \exists Pos(r, x + 1, y) \}$
- e3. $\forall r, x, y, u, v R(r, S), Pos^*(r, x, y), Free^\#(x, y - 1), App(r, u, v), dist(x, y, u, v) \geq dist(x, y - 1, u, v) \{ \exists Pos(r, x, y - 1) \}$
- e4. $\forall r, x, y, u, v R(r, W), Pos^*(r, x, y), Free^\#(x - 1, y), App(r, u, v), dist(x, y, u, v) \geq dist(x - 1, y, u, v) \{ \exists Pos(r, x - 1, y) \}$

Если робот находится рядом с блоком и готов к его перемещению, то ему необходимо повернуться на север:

$$\forall r, x, y, d R^*(r, d), Pos(r, x, y - 1), App^*(r, x, y) \{ \exists R(r, N), Ready(r) \}$$

Для проверки достижения второго пункта назначения зададим также два правила: необходимо повернуть в нужном направлении, если в базе присутствует предикат $BlockMoved(b, n)$:

1. $\forall r, b, n, x, y BlockMoved(b, n), Pos(b, x, y), App^*(r, x - 1, y) \{ \exists Ready(r), R(r, E) \}$
2. $\forall r, b, n, x, y BlockMoved(b, n), Pos(b, x, y), App^*(r, x + 3, y) \{ \exists Ready(r), R(r, W) \}$

Для АТР на основе ПОФ-исчисления разработан пружер Bootfrost, который специализируется на выводе ПОФ без неограниченных переменных (<https://github.com/snigavik/bootfrost>). Пружер был протестирован на всемирно известной библиотеке задач для

тестирования прuverов ТРТР. Ни одна из задач, не имевших решения, не была решена, что считается основным критерием надежности любого прuverа на крупнейших соревнованиях прuverов CADE ATP System Competition (<https://tptp.org/CASC/>). В системе управления, в подсистеме символьной обработки, система логического вывода представляется параллельным потоком с высшим приоритетом. Поток реализует прuver Vooftrost и обеспечивает переключение алгоритмов ситуационного уровня («задач») в соответствии с полученными результатами.

Выводы

В работе предложен иерархический подход для динамического управления разнородной группой АМР в задачах групповой координации в условиях жестких коммуникационных и энергетических ограничений. Предлагаемая схема имеет трехуровневую структуру, где верхний уровень отвечает за планирование рабочих периодов группы, сеансов групповой связи и расписания подзарядки роботов, средний уровень обеспечивает управление АМР при выполнении текущих задач, а нижний уровень служит для обработки динамических событий и связанного с ними переключения режимов функционирования. Такая декомпозиция на три уровня позволяет разгрузить общую проблему распределения заданий и ресурсов в группе, выделив наиболее жесткие ограничения в отдельную проблему управления временем.

Представленная в работе задача планирования высокого уровня заключается в построении группового расписания, обеспечивающего эффективный порядок подзарядки и требуемую периодичность связи. Задача формулируется в терминах задач планирования рабочих смен, а для ее решения разработана гибридная модификация эволюционных алгоритмов.

На уровне логической обработки данных используется метод автоматического доказательства теорем в исчислении ПОФ, разработанного для управления динамическими интеллектуальными системами. Наглядное представление ПОФ, интуитивно понятная формализация задач, а также ряд важных особенностей исчисления делают его эффективным инструментом интеллектуализации систем управления АМР. Представлен пример применения логического вывода для генерации событий смены режимов функционирования АМР. Специальные логические правила, представленные в ПОФ в виде вопросов обработки событий, запускают подвыводы, в которых поступающие данные служат параметрами, используемыми в расчетах и других процессах принятия решений. Особенности исчисления, связанные с высокоуровневой обработкой событий и возможностью накопления знаний, будут полезны в задачах, возникающих в реальных системах, функционирующих в заранее неизвестных средах. Строгое формальное обоснование полученных в результате логического вывода результатов гарантирует корректность принимаемых решений.

Дальнейшие исследования будут направлены на решение задач управления для децентрализованных и распределенных систем, а также на внедрение представленного в статье подхода в современные робототехнические комплексы.

Список литературы

1. Badreldin M, Hussein A and Khamis A. A comparative study between optimization and market-based approaches to multi-robot task allocation // *Advances in Artificial Intelligence*. 2013. 2013(256524).
2. Vidal T, Crainic TG, Gendreau M et al. Heuristics for multi-attribute vehicle routing problems: A survey and synthesis // *European Journal of Operations Research*. 2012. Vol 231. P. 1-21.
3. Gini M. Multi-robot allocation of tasks with temporal and ordering constraints // *Proceedings of 31st AAAI Conference on Artificial Intelligence, AAAI 2017*. 2017. P. 4863-4869.

4. Nunes E, Manner M, Mitiche H et al. A taxonomy for task allocation problems with temporal and ordering constraints // *Robotics and Autonomous Systems Special Issue on New Research Frontiers for Intelligent Autonomous Systems*. 2017. Vol. 90. P. 55-70.
5. Stephan J, Fink J, Kumar V et al. Concurrent control of mobility and communication in multirobot systems // *IEEE Transactions on Robotics*. 2017. Vol. 33(5). P. 1248-1254.
6. Varadharajan V, St-Onge D, Adams B et al. Swarm relays: distributed self-healing ground-and-air connectivity chains // *IEEE Robotics and Automation Letters*. 2020.
7. Kantaros Y, Guo M and Zavlanos M. Temporal logic task planning and intermittent connectivity control of mobile robot networks // *IEEE Transactions on Automatic Control*. 2019. Vol. 64(10). P. 4105-4120.
8. Asghari M and Mirzapour Al-e-hashem SMJ. Green vehicle routing problem: A state-of-the-art review // *International Journal of Production Economics*. 2021. Vol. 231: 107899.
9. Zou B, Xu X, Gong Y et al. Evaluating battery charging and swapping strategies in a robotic mobile fulfillment system // *European Journal of Operation Research*. 2018. Vol. 267(2). P. 733-753.
10. MahmoudZadeh S, Powers DMW, Sammut K et al. Hybrid motion planning task allocation model for AUV's safe maneuvering in a realistic ocean environment // *Journal of Intelligent & Robotic Systems*. 2019. Vol. 94(1). P. 265–282.
11. Karpas, E.; Magazzeni, D. Automated planning for robotics. *Annual Review of Control, Robotics, and Autonomous Systems* 2020, 3, 417–439.
12. Zombori, Z.; Urban, J.; Brown, C.E. Prolog technology reinforcement learning prover. In *Proceedings of the International Joint Conference on Automated Reasoning*. Springer, 2020, pp. 489–507.
13. Schader, M.; Luke, S. Planner-Guided Robot Swarms. In *Proceedings of the International Conference on Practical Applications of Agents and Multi-Agent Systems*. Springer, 2020, pp. 224–237.
14. Vassilyev S.N. Machine Synthesis of Mathematical Theorems // *The Journal of Logic programming*. 1990. Vol. 9, No. 2-3, P. 235-266.
15. Васильев С.Н. Интеллектуальное управление динамическими системами / С.Н. Васильев, А.К. Жерлов, Е.А. Федунцов, Б.Е. Федосов. – М.: Физико-математическая литература, 2000. – 352с.
16. Davydov A.V., Larionov A.A., Cherkashin E.A. On the calculus of positively constructed formulas for automated theorem proving // *Automatic Control and Computer Sciences (AC&CS)*. 2011. Vol. 45, No. 7, P. 402-407.
17. Cherkashin, E.A.; Postoenko, A.; Vassilyev, S.N.; Zherlov, A. New Logics for Intelligent Control. In *Proceedings of the Proceedings of the Twelfth International Florida Artificial Intelligence Research Society Conference, May 1-5, 1999, Orlando, Florida, USA*; Kumar, A.N.; Russell, I., Eds. AAAI Press, 1999, pp. 257–261.
18. Vassilyev S., Galyaev A. Logical-optimization approach to pursuit problems for a group of targets. *Dokl. Math*. 2017, 95.
19. Vassilyev S., Ponomarev G. Automation methods for logical derivation and their application in the control of dynamic and intelligent systems. *Proc. Steklov Inst. Math*. 2012, 276, 161–179.
20. Schacht-Rodríguez R, Ponsart J.-C., García-Beltrán C.D. and Astorga-Zaragoza C.M. Prognosis & Health Management for the prediction of UAV flight endurance // *IFAC-PapersOnLine*. 2018. Vol. 51(24). P. 983–990.
21. Van den Bergh J, Beliën J, De Bruecker P et al. Personnel scheduling: A literature review // *European Journal of Operational Research*. 2013. Vol. 226(3). P. 367-385.
22. Rocha M. The staff scheduling problem: a general model and applications // *Master thesis: Faculdade de Engenharia da Universidade do Porto*. 2013.
23. Chan P, Weil G. Cyclical Staff Scheduling Using Constraint Logic Programming // *Lecture Notes in Computer Science*. 2000. Vol. 2079. P. 159–175.
24. Amjad MK, Butt SI, Kousar R et al. Recent research trends in genetic algorithm based flexible job shop scheduling problems // *Mathematical Problems in Engineering*. 2018. Vol. 2018(5). P. 1-32.
25. Vidal T, Crainic TG, Gendreau M et al. A hybrid genetic algorithm for multidepot and periodic vehicle routing problems // *Operations Research*. 2012 Vol. 60(3). P. 611-624.
26. Laporte G, Ropke S, Vidal T. Chapter 4: Heuristics for the Vehicle Routing Problem // *Vehicle Routing*. 2014. P. 87–116.
27. Nagata Y, Bräysy O, Dullaert W. A penalty-based edge assembly memetic algorithm for the vehicle routing problem with time windows // *Computers & Operations Research*. 2010. Vol. 37(4). P. 724–737.
28. Semanco P and Modrak V. A comparison of constructive heuristics with the objective of minimizing makespan in the flow-shop scheduling problem // *Acta Polytechnica Hungarica*. 2012. Vol. 9. P. 177-190.
29. Laporte G. The art and science of designing rotating schedules // *European Journal of Operational Research*. 1999. Vol. 50. P. 1011-1017.
30. Cassandras C.G., Lafortune S. *Introduction to Discrete Event Systems*. Springer Cham, 2021.
31. Давыдов А. В. О применении исчисления позитивно-образованных формул для исследования управляемых дискретно-событийных систем / А. В. Давыдов, А. А. Ларионов, Н. В. Нагул. *Модели и анализ информ. систем*. 2024. № 1, Т. 31. С. 54–77.

References

1. Badreldin M, Hussein A and Khamis A. A comparative study between optimization and market-based approaches to multi-robot task allocation // *Advances in Artificial Intelligence*. 2013. 2013(256524).
2. Vidal T, Crainic TG, Gendreau M et al. Heuristics for multi-attribute vehicle routing problems: A survey and synthesis // *European Journal of Operations Research*. 2012. Vol 231. P. 1-21.
3. Gini M. Multi-robot allocation of tasks with temporal and ordering constraints // *Proceedings of 31st AAAI Conference on Artificial Intelligence, AAAI 2017*. 2017. P. 4863--4869.
4. Nunes E, Manner M, Mitiche H et al. A taxonomy for task allocation problems with temporal and ordering constraints // *Robotics and Autonomous Systems Special Issue on New Research Frontiers for Intelligent Autonomous Systems*. 2017. Vol. 90. P. 55-70.
5. Stephan J, Fink J, Kumar V et al. Concurrent control of mobility and communication in multirobot systems // *IEEE Transactions on Robotics*. 2017. Vol. 33(5). P. 1248-1254.
6. Varadharajan V, St-Onge D, Adams B et al. Swarm relays: distributed self-healing ground-and-air connectivity chains // *IEEE Robotics and Automation Letters*. 2020.
7. Kantaros Y, Guo M and Zavlanos M. Temporal logic task planning and intermittent connectivity control of mobile robot networks // *IEEE Transactions on Automatic Control*. 2019. Vol. 64(10). P. 4105-4120.
8. Asghari M and Mirzapour Al-e-hashem SMJ. Green vehicle routing problem: A state-of-the-art review // *International Journal of Production Economics*. 2021. Vol. 231: 107899.
9. Zou B, Xu X, Gong Y et al. Evaluating battery charging and swapping strategies in a robotic mobile fulfillment system // *European Journal of Operation Research*. 2018. Vol. 267(2). P. 733-753.
10. MahmoudZadeh S, Powers DMW, Sammut K et al. Hybrid motion planning task allocation model for AUV's safe maneuvering in a realistic ocean environment // *Journal of Intelligent & Robotic Systems*. 2019. Vol. 94(1). P. 265-282.
11. Karpas, E.; Magazzeni, D. Automated planning for robotics. *Annual Review of Control, Robotics, and Autonomous Systems* 2020, 3, 417-439.
12. Zombori, Z.; Urban, J.; Brown, C.E. Prolog technology reinforcement learning prover. In *Proceedings of the International Joint Conference on Automated Reasoning*. Springer, 2020, pp. 489-507.
13. Schader, M.; Luke, S. Planner-Guided Robot Swarms. In *Proceedings of the International Conference on Practical Applications of Agents and Multi-Agent Systems*. Springer, 2020, pp. 224-237.
14. Vassilyev S.N. Machine Synthesis of Mathematical Theorems // *The Journal of Logic programming*. 1990. Vol. 9, No. 2-3, P. 235-266.
15. Vassilyev S. N. Intelligent control of dynamic systems / A. K. Zherlov, S. N. Vassilyev, E. A. Fedosov, and B. E. Fedunov, *Fizmatlit*, 2000, 351 pp.
16. Davydov A.V., Larionov A.A., Cherkashin E.A. On the calculus of positively constructed formulas for automated theorem proving // *Automatic Control and Computer Sciences (AC&CS)*. 2011. Vol. 45, No. 7, P. 402-407.
17. Cherkashin, E.A.; Postoenko, A.; Vassilyev, S.N.; Zherlov, A. New Logics for Intelligent Control. In *Proceedings of the Proceedings of the Twelfth International Florida Artificial Intelligence Research Society Conference, May 1-5, 1999, Orlando, Florida, USA*; Kumar, A.N.; Russell, I., Eds. AAAI Press, 1999, pp. 257-261.
18. Vassilyev S., Galyaev A. Logical-optimization approach to pursuit problems for a group of targets. *Dokl. Math*. 2017, 95.
19. Vassilyev S., Ponomarev G. Automation methods for logical derivation and their application in the control of dynamic and intelligent systems. *Proc. Steklov Inst. Math*. 2012, 276, 161-179. pas, E.; Magazzeni, D. Automated planning for robotics. *Annual Review of Control, Robotics, and Autonomous Systems* 2020, 3, 417-439.
20. Schacht-Rodríguez R, Ponsart J.-C., García-Beltrán C.D. and Astorga-Zaragoza C.M. Prognosis & Health Management for the prediction of UAV flight endurance // *IFAC-PapersOnLine*. 2018. Vol. 51(24). P. 983-990.
21. Van den Bergh J, Beliën J, De Bruecker P et al. Personnel scheduling: A literature review // *European Journal of Operational Research*. 2013. Vol. 226(3). P. 367-385.
22. Rocha M. The staff scheduling problem: a general model and applications // *Master thesis: Faculdade de Engenharia da Universidade do Porto*. 2013.
23. Chan P, Weil G. Cyclical Staff Scheduling Using Constraint Logic Programming // *Lecture Notes in Computer Science*. 2000. Vol. 2079. P. 159-175.
24. Amjad MK, Butt SI, Kousar R et al. Recent research trends in genetic algorithm based flexible job shop scheduling problems // *Mathematical Problems in Engineering*. 2018. Vol. 2018(5). P. 1-32.
25. Vidal T, Crainic TG, Gendreau M et al. A hybrid genetic algorithm for multidepot and periodic vehicle routing problems // *Operations Research*. 2012 Vol. 60(3). P. 611-624.
26. Laporte G, Ropke S, Vidal T. Chapter 4: Heuristics for the Vehicle Routing Problem // *Vehicle Routing*. 2014. P. 87-116.
27. Nagata Y, Bräysy O, Dullaert W. A penalty-based edge assembly memetic algorithm for the vehicle routing problem with time windows // *Computers & Operations Research*. 2010. Vol. 37(4). P. 724-737.
28. Semanco P and Modrak V. A comparison of constructive heuristics with the objective of minimizing makespan in the flow-shop scheduling problem // *Acta Polytechnica Hungarica*. 2012. Vol. 9. P. 177-190.

29. Laporte G. The art and science of designing rotating schedules // European Journal of Operational Research. 1999. Vol. 50. P. 1011-1017.
30. Cassandras C.G., Lafortune S. Introduction to Discrete Event Systems. Springer Cham, 2021.
31. Davydov A. V. On the application of the calculus of positively constructed formulas for the study of controlled discrete-event systems / A. V. Davydov, A. A. Larionov, N. V. Nagul // Model. Anal. Inform. Sist. 2024. № 1, V. 31. P. 54–77.

RESUME

I. V. Bychkov, A. V. Davydov, M. Yu. Kenzin, N. V. Nagul
Hierarchical Planning of Actions of a Heterogeneous Group
of Autonomous Mobile Robots

The use of a group of autonomous mobile robots (AMRs) requires solving a number of new complex problems, since the quality of the group's work in a dynamic environment depends to a greater extent on the effectiveness of intra-group interaction schemes than on the reliability of individual team members. In general, the mission of the AMR team is to perform a certain set of tasks within a designated area in accordance with current operational requirements. A review of the literature shows that communication and energy limitations can be identified as group-wide restrictions, the work with which requires a special approach. Despite the huge number of approaches that take into account each of these two limitations, there is almost no work in the literature in which they are used simultaneously and applied to a functionally heterogeneous group of robots.

The article proposes a hierarchical approach for the dynamic control of a heterogeneous group of AMRs in group coordination problems under severe communication and energy constraints. The proposed scheme has a three-level structure, where the top level is responsible for planning group work periods, group communication sessions and robot recharging schedules, the middle level provides control of the AMR when performing current tasks, and the lower level is used for processing dynamic events and associated switching of operating modes. Decomposition into three levels allows one to unload the general problem of distributing tasks and resources in a group, highlighting the most stringent restrictions into a separate problem of time management.

The task of the planner at the top level is to ensure effective rotation of the AMR group throughout the mission in dynamic conditions and taking into account the requirements for regular communication by constructing an effective schedule of work periods. To solve the problem, a problem-oriented modification of evolutionary algorithms is proposed, enriched with several specialized heuristics and improvement operators. The event handler level, responsible for switching the operating modes of the AMR based on emerging events, is implemented based on the formalism of logical discrete event systems (DESs) in the form of finite state automata. The set of sequences of events forms a formal language that describes the functioning of the system at the symbolic level, and statements about the properties of this language are proven using automatic theorem proving (ATP) in the calculus of positively constructed formulas (PCFs). It is shown how the ATP in the PCF calculus is used to analyze and synthesize the control of logical DESs.

РЕЗЮМЕ

И. В. Бычков, А. В. Давыдов, М. Ю. Кензин, Н. В. Нагул
Иерархическое планирование действий разнородной группы
автономных мобильных роботов

Применение группы автономных мобильных роботов (АМР) требует решения ряда новых комплексных задач, так как качество работы группы в динамичной среде в большей степени зависит от эффективности схем внутригруппового взаимодействия, чем от надежности отдельных членов команды. В общем случае миссия группы АМР заключается в выполнении некоторого набора заданий в пределах обозначен-

ной области в соответствии с текущими эксплуатационными требованиями. Обзор литературы показывает, что в качестве общегрупповых ограничений, работа с которыми требует особого подхода, могут быть выделены коммуникационные и энергетические ограничения. Несмотря на огромное количество подходов, учитывающих каждое из этих двух ограничений, в литературе почти не встречается работ, в которой они использовались бы одновременно и применялись к функционально гетерогенной группе роботов.

В статье предложен иерархический подход для динамического управления разнородной группой АМР в задачах групповой координации в условиях жестких коммуникационных и энергетических ограничений. Предлагаемая схема имеет трехуровневую структуру, где верхний уровень отвечает за планирование рабочих периодов группы, сеансов групповой связи и расписания подзарядки роботов, средний уровень обеспечивает управление АМР при выполнении текущих задач, а нижний уровень служит для обработки динамических событий и связанного с ними переключения режимов функционирования. Декомпозиция на три уровня позволяет разгрузить общую проблему распределения заданий и ресурсов в группе, выделив наиболее жесткие ограничения в отдельную проблему управления временем.

Задача планировщика на верхнем уровне – обеспечить эффективную ротацию АМР группы на протяжении всей миссии в динамических условиях и с учетом требований к регулярности связи путем построения эффективного расписания рабочих периодов. Для решения поставленной задачи предложена проблемно-ориентированная модификация эволюционных алгоритмов, обогащенная несколькими специализированными эвристиками и операторами улучшения. Уровень обработчика событий, отвечающий за переключение режимов функционирования АМР на основе возникающих событий, реализован на основе формализма логических дискретно-событийных систем (ДСС) в форме конечных автоматов. Множество последовательностей событий образует формальный язык, описывающий функционирование системы на символическом уровне, а утверждения о свойствах этого языка доказываются с помощью автоматического доказательства теорем (АДТ) в исчисления позитивно-образованных формул (ПОФ). Показано, как АДТ в исчислении ПОФ используется для анализа и синтеза управления логическими ДСС.

Бычков И. В. – академик, д.т.н., директор ИДСТУ СО РАН им. В.М. Матросова, 664033, Иркутск, ул. Лермонтова, 134, тел. +7(395) 242-7100, bychkov@icc.ru.

Область научных интересов: комплексные информационные сети, распределенные вычисления и технологии, искусственный интеллект

Давыдов А. В. – научный сотрудник ИДСТУ СО РАН им. В.М. Матросова, лаборатория информационно-управляющих систем, 664033, Иркутск, ул. Лермонтова, 134, тел. +7(395) 245-3085, artem@icc.ru. *Область научных интересов:* математическая логика, искусственный интеллект, робототехника

Кензин М. Ю. – научный сотрудник ИДСТУ СО РАН им. В.М. Матросова, лаборатория информационно-управляющих систем, 664033, Иркутск, ул. Лермонтова, 134, тел. +7(395) 245-3085, gorthauers@gmail.com.

Область научных интересов: метаэвристические алгоритмы, имитационное моделирование, мультиагентные системы

Нагул Н. В. – к.ф.-м.н., ИДСТУ СО РАН им. В.М. Матросова, с.н.с., заведующая лабораторией информационно-управляющих систем, 664033, Иркутск, ул. Лермонтова, 134, тел. +7(395) 245-3085, sapling@icc.ru. *Область научных интересов:* системный анализ, теория управления, дискретно-событийные системы

Статья поступила в редакцию 14.02.2024.