

ДО ПОСТАНОВКИ ЗАДАЧІ РОЗПІЗНАВАННЯ НЕВІДОМОГО ОТОЧУЮЧОГО СЕРЕДОВИЩА, НАВІГАЦІЇ ТА ПЛАНУВАННЯ ШЛЯХІВ АГЕНТОМ В НЬОМУ

Наведено аналітичний огляд основних тенденцій, домінуючих в світі у рамках вирішення проблеми розпізнавання невідомого оточуючого середовища, навігації та планування шляхів агентом у ньому. Проаналізовано структуру задач, які є складовими цієї проблеми, та визначено, що напрям пропонуваніх досліджень належить до проблематики одночасної побудови мапи та планування шляху. На основі здійсненого огляду виконано постановку задач досліджень для випадку одного агента та визначено перелік методів, що потребують першочергової розробки.

Ключові слова: агент, невідоме оточуюче середовище, мапа, локалізація, планування шляхів, траєкторія руху, граф пошуку.

Вступ

Проблема планування дій мобільного агента (прототипу мобільного робота) в невідомому оточуючому середовищі на сьогоднішній день є однією з найбільш складних та актуальних задач у галузі штучного інтелекту: очікується, що її вирішення створить умови для потужного розвитку та удосконалення відповідних технологій робототехніки. Відомо, що мобільні роботи можуть бути застосовані не тільки для вирішення побутових завдань (наприклад, для автоматичного прибирання приміщень (пилосос)), а й для виконання спеціальних завдань (у тому числі пошуково-рятувальних та розвідувальних дій) у невідомому та небезпечному оточуючому середовищі. Водночас, реальне використання мобільних роботів для відпрацювання відповідних методів пов'язано з певними ускладненнями і в загальному випадку є неефективним: по-перше – це коштовний та трудомісткий процес; по-друге, на практиці таке використання відбувається в конкретних фіксованих умовах оточуючого середовища, що не дозволяє гарантувати робастність відповідних методів у будь-якій ситуації для довільного оточуючого середовища.

На відміну від цього, використання мультиагентного підходу для вирішення означених задач дозволяє перевірити методи моделювання поведінки агентів у будь-яких ситуаціях для множини оточуючих середовищ з різними властивостями.

Тому далі ми будемо вживати поняття «агент», розуміючи під ним прототип мобільного робота.

Вирішення досліджуваної проблеми для випадку одного агента передбачає розв'язання трьох взаємопов'язаних задач його навігації: *побудови мапи* (mapping), *локалізації* (localization) та *планування шляху* (path planning). Побудова мапи – це проблема інтеграції інформації, що надходить з сенсорів агента, зібраної з різних місць його розташування в оточуючому середовищі. Коректна обробка даних сенсорів і точна локалізація є фундаментальними для забезпечення адекватності створюваної мапи. Локалізація – це проблема оцінювання фактичного місцезнаходження агента, виходячи з показників сенсорів та відомостей з вже побудованої мапи. Точна мапа та надійні сенсори мають вирішальне значення для визначення правильної локалізації. Планування шляху або проблема керування рухом – це проблема створення допустимої та безпечної (що унеможливає зіткнення з будь-якими зустрічними об'єктами) траєкторії руху від поточного місцезнаходження агента до цілі, де створена траєкторія має враховувати обмеження, що містяться на вже існуючій мапі. При цьому важливе значення мають точність побудови мапи та надійність визначення локалізації.

На рис. 1 показано три зазначені задачі та їх перетини, що формують основні

сучасні напрями досліджень у рамках вирішення аналізованої проблеми.

Так, перетин задач побудови мапи та локалізації утворив напрям досліджень з *одночасної локалізації та побудови мапи* (simultaneous localization and mapping (SLAM)). Ці задачі є взаємопов'язаними та взаємозалежними і тому SLAM часто називають проблемою «куриці» та «яйця»: для рішення задач локалізації потрібна точна мапа, тоді як для побудови мапи потрібна точна оцінка місцезнаходження агента. Основні аспекти сучасного стану досліджень у галузі SLAM викладено в [3, 4].

Активна локалізація спрямована на те, щоб спланувати дії агента щодо знаходження такого місця на мапі, яке дозволить покращити оцінку його місцезнаходження. Зазначимо, що більшість існуючих підходів до локалізації є пасивними, тобто орієнтованими на оцінку місцезнаходження агента виключно на основі даних, що надходять від сенсорів. Активна локалізація додатково передбачає, що процедура локалізації має частковий або повний контроль над діями агента, надаючи можливість покращити ефективність та надійність локалізації. При цьому ключовими проблемами є вирішення питань «куди рухатись» та «де шукати», щоб найкращим чином визначити місцезнаходження агента.

В свою чергу, перетин задач побудови мапи та планування руху утворює наш напрям досліджень. Відомі підходи у цьому напрямку досліджень передбачають наявність точної інформації про місцезнаходження агента та спрямовані на вирішення задач ефективного керування його рухом через навколишнє середовище з метою побудови мапи. Ці підходи, як правило, використовують такі подання мапи як подання, засновані на орієнтирах та подання графу огляду (див. п. 2 статті), і задача планування шляхів відповідає задачі локального пошуку шляху. В нашому випадку теж передбачається наявність точної інформації про місцезнаходження агента, але, на відміну від відомих підходів, ми плануємо вирішувати проблему побудови мапи шляхом розпізнавання і класифікації агентом об'єктів оточуючого середовища в процесі його руху через оточуюче середовище з метою одночасного плануванням агентом свого подальшого шляху за допомогою методів локального пошуку. Якщо відомі підходи можна узагальнено назвати як підходи з *одночасного планування шляху та побудови мапи* (simultaneous planning and mapping (SPAM)), то наш напрям досліджень – підходом з *одночасної побудови мапи та планування шляху* (simultaneous mapping and planning (SMAP)).

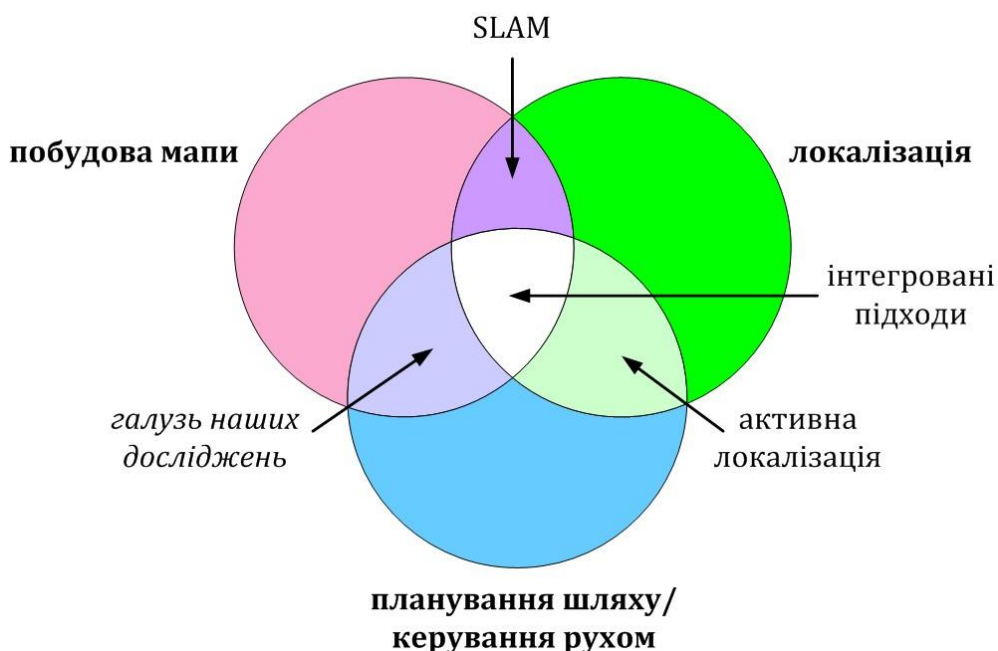


Рис.1. Галузі досліджень проблеми навігації агентів (адаптовано з [1, 2])

Зазначимо також, що підходи з подання мапи, що включають подання, засновані на зайнятості, геометричні подання та подання графа маршруту (див. п. 2 статті), орієнтовані на використання глобальних методів пошуку на дискретному поданні (у вигляді дорожньої мапи, подання, заснованому на клітковій декомпозиції тощо) вільного простору оточуючого середовища агента. Тобто такі підходи передбачають існування мапи оточуючого середовища та перетворення її у відповідне дискретне подання перед виконанням пошуку шляхів і використовуються, зокрема, в рамках напряму досліджень з планування шляху/керування рухом (див. рис. 1).

Перетин всіх трьох задач утворює напрям досліджень, що відповідає так званим *інтегрованим підходам*, які одночасно торкаються розв'язування проблем побудови мапи, локалізації та планування шляху. Інтегровані підходи також називаються підходами з рішення проблеми *одночасного планування, локалізації та побудови мапи* (simultaneous planning, localization, and mapping (SPLAM)). Рішення проблеми SPLAM дозволяє агенту отримувати дані датчиків, автономно рухаючись через своє оточуюче середовище, та одночасно будувати мапу. В кожний момент часу свого руху агент обирає дії, спрямовані на поліпшення своєї локалізації, отримує інформацію про фрагменти невідомого оточуючого середовища та поліпшує модель своєї мапи шляхом їх обробки. Нарешті передбачається, що він має створити точну модель всього оточуючого середовища, а також визначити своє місцезнаходження в межах цієї моделі.

На сьогоднішній день світові дослідження охоплюють всі названі напрями. При цьому аналізуються різні аспекти проблеми, для чого досліджується поведінка як одного агента, так і групи агентів. Зважаючи на складність аналізованої проблеми, використання для її вирішення групи агентів (у порівнянні з використанням одного агента) має як свої переваги, так і недоліки. До переваг, зокрема, належать наступні: агенти, що кооперуються, спроможні швидше вирішити відповідні задачі (наприклад, побудову загальної

мапи, знаходження шуканих об'єктів тощо), ніж один агент; у групи агентів більша імовірність успішно завершити вирішення покладених на них завдань, ніж у одного агента, оскільки відмова окремого агента групи не унеможливорює отримання загального кінцевого результату. Одним з основних недоліків, що може негативно вплинути на ефективність вирішення задач, є наступний: в процесі руху агентів, що належать групі, можливі ситуації їх зустрічей, внаслідок чого вони мають вирішувати задачі уникнення зіткнень шляхом затримок або об'їздів таких перешкод, що збільшує час вирішення загальної задачі. Тобто на відміну від ситуації з одним агентом, коли, наприклад, його оточуюче середовище є статичним, у випадку розгляду групи агентів це оточуюче середовище стає динамічним за рахунок динамічної зміни станів агентів, що входять до складу групи.

Зазначимо, що в наших дослідженнях планується вирішувати проблему SMAP за допомогою групи агентів, які мають координувати свої дії та кооперуватися між собою для вирішення загальної задачі.

1. Сутність проблеми локалізації

Вирішення проблеми локалізації знаходиться в безпосередньому зв'язку з наступними аспектами її розгляду [5]:

- 1) ступінь інформованості агента щодо його поточного розташування;
- 2) тип оточуючого середовища, в якому діє агент;
- 3) ступінь контролю за рухом агента з боку процедури локалізації;
- 4) кількість агентів, що одночасно діють в оточуючому середовищі.

Коротко охарактеризуємо сутність проблеми локалізації з точки зору кожного із зазначених аспектів.

Ступінь інформованості агента щодо його поточного розташування залежить від точності та повноти інформації, якою він оперує у процесі руху. Розрізняють три можливі ситуації:

- локальна локалізація (local localization) або відстеження місцезнахо-

дження (position tracking) характеризується тим, що агент відстежує своє місцезнаходження у кожний момент часу, виходячи з відомих координат його розташування на початку руху. Вважається, що в агента похибки оцінки переміщення (одометрія) та кутів поворотів незначні. Це найпростіша ситуація щодо вирішення проблеми локалізації;

- глобальна локалізація (global localization) характеризується тим, що агенту невідомі координати його розташування на початку руху, і він має відстежувати своє поточне місцезнаходження, перебуваючи в межах оточуючого середовища, але не знаючи, де саме він знаходиться. Ця ситуація складніша, ніж попередня;

- проблема «викрадення» (kidnapped problem) агента, яка є найбільш складним випадком глобальної локалізації. Сутність проблеми полягає у тому, що агент був без його відома переміщений в інше місце, але він впевнений при цьому, що своє поточне місцезнаходження визначає коректно. Вирішення цієї проблеми (забезпечення коректного визначення місцезнаходження) націлене на підвищення спроможності агента відновлюватись після помилок глобальної локалізації.

Тип оточуючого середовища, в якому діє агент, також може впливати на коректність вирішення проблеми локалізації. Оточуюче середовище є статичним, якщо в ньому рухається лише один агент, а інші об'єкти не змінюють свого геометричного розташування. На відміну від цього, в динамічному оточуючому середовищі можуть рухатись й інші об'єкти, розташування яких може постійно змінюватись, тим самим впливаючи на коректність вирішення проблеми локалізації.

За ступенем контролю за рухом агента з боку процедури локалізації розрізняють випадки пасивної та активної локалізації (див. рис. 1). У випадку пасивної локалізації визначення місцезнаходження агента відбувається тільки на основі показників сенсорів. У випадку активної локалізації відповідні процедури керують рухом агента з метою мінімізації помилок локалізації.

В залежності від кількості агентів, що одночасно діють в оточуючому середовищі, проблема локалізації вирішується або для одного агента, або для множини агентів. У першому випадку проблема локалізації стосується тільки одного агента і не ускладнена необхідністю одночасного вирішення проблеми комунікації з іншими агентами. У другому випадку агенти повинні кооперуватись між собою з метою побудови загальної мапи, і некоректне вирішення проблеми локалізації може суттєво вплинути на якість створення такої мапи, оскільки одні і ті ж самі об'єкти можуть мати різні координати на мапі внаслідок похибок вимірювання, отриманих різними агентами.

Для вирішення проблеми локалізації використовуються різні підходи, в тому числі й імовірнісні, які ґрунтуються на використанні фільтрів Байєса (зокрема, алгоритм локалізації Маркова, алгоритм локалізації Монте Карло, алгоритм фільтру Калмана та його модифікації тощо). Детально різні аспекти проблеми локалізації викладено в [5, 6].

2. Огляд відомих підходів до побудови та подання мапи оточуючого середовища

Побудова агентом мапи оточуючого середовища – це складна науково-практична проблема, вирішення якої передбачає спроможність агента, починаючи з деякої початкової точки його розташування, автономно дослідити оточуюче середовище за допомогою своїх сенсорів, та отримати й інтерпретувати відповідні дані, на основі яких створити власну мапу та локалізувати себе відносно її. При побудові мапи використовуються відповідні підходи з її подання.

На рис. 2 на верхньому рівні таксономії класифікацію підходів виконано виходячи з розбіжностей у поданні просторових відношень між основними сутностями мапи, що дозволило виділити два основних класи таксономії: подання, засновані на координатах (або метричні подання) та реляційні подання (які, зокрема,

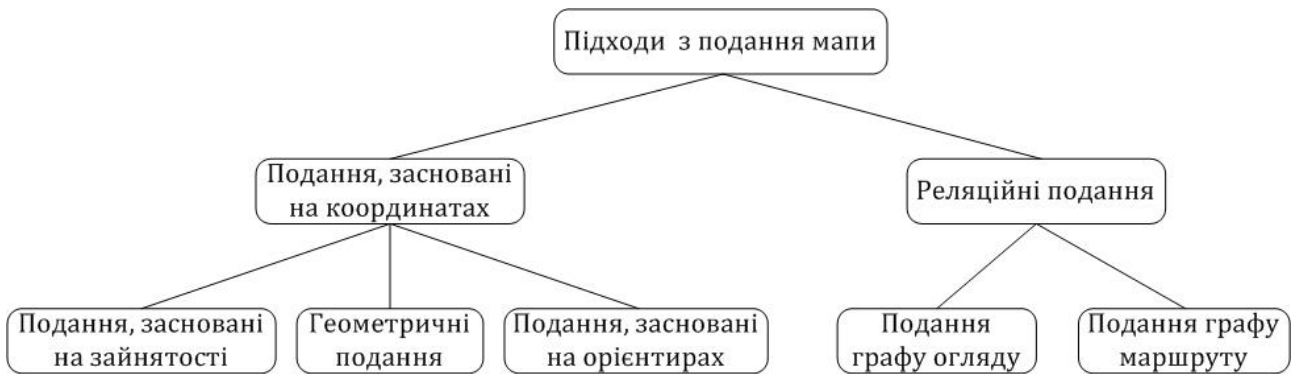


Рис. 2. Таксономія існуючих підходів з подання мапи (адаптовано з [7])

включають топологічні мапи). Розбіжності між цими поданнями полягають у наступному:

- подання, засновані на координатах, неявно відображають просторові відношення між основними сутностями, надаючи координати для кожного з просторових об'єктів у рамках єдиної глобальної системи координат;

- реляційні подання виражають просторові відношення між основними сутностями, явно визначаючи, що існує відповідне відношення між відповідним набором об'єктів.

Подання, засновані на координатах, поділяються на три підкласи: *подання, засновані на зайнятості*; *геометричні подання* та *подання, засновані на орієнтирах*.

Подання, засновані на зайнятості (Occupancy-based representations) орієнтовані на визначення зайнятих та вільних частин простору шляхом його розбиття на зайняті та вільні клітинки. Як правило, розбиття не залежить від розподілу об'єктів у просторі та є рівномірним в тому сенсі, що всі клітинки мають однакову форму і розмір. При цьому в більшості випадків використовується так звана мережа зайнятості (occupancy grid), в якій подається невизначеність щодо стану зайнятості кожної клітинки як значення правдоподібності $l \in [0,1]$. Переваги використання мереж зайнятості полягають у легкості злиття та гнучкості при об'єднанні даних, отриманих агентом з різних типів сенсорів. Недоліки їх використання полягають у значних обчислювальних витратах при обробці великих мереж зайнятості та труднощах у

знаходженні оптимальних шляхів на мапі без залучення людини.

Геометричні подання (Geometric representations) використовують множину геометричних примітивів (точки, лінії, криві тощо) для опису множини перешкод як границь вільного простору, розташованих в єдиній системі координат. Геометричні подання дозволяють якісно подавати довільні середовища за умови використання геометричних примітивів, які забезпечують коректну апроксимацію форм границь об'єкта (наприклад, точок, ліній, поліліній). Таке подання, як правило, є компактним. Але при цьому існує проблема злиття даних для об'єднання відповідних об'єктів на мапі, яка залишається однією з найбільш складних проблем у рамках геометричного подання. Оскільки геометричні подання описують границі вільного простору середовища, вони, з одного боку, добре підходять для вирішення задач локалізації, з іншого, – для планування шляхів. Водночас планування шляхів вимагає створення дискретного простору пошуку в межах середовища для використання відомих методів пошуку, що призводить до додаткових обчислювальних витрат. Типовими підходами для вирішення таких задач є підходи дорожньої мапи (roadmap), кліткової декомпозиції (cell decomposition) тощо. До недоліків геометричного подання можна віднести необхідність постійного відстеження, які частини оточуючого середовища вже оглянуті та потребують явного подання і постійного поновлення опису оглянутої області.

Подання, засновані на орієнтирах (Landmark-based representations) подають

оточуюче середовище як набір важливих об'єктів (орієнтирів), видобутих з даних сенсорів, де позиції орієнтирів визначаються у глобальній системі координат. Геометричні подання та подання, засновані на орієнтирах, належать до *подання, заснованого на характеристиках* (feature-based representations), оскільки ці підходи збігаються за ознакою подібності структур даних про об'єкти оточуючого середовища та використовуваних методів обробки невизначеності. Однак між цими поданнями існують розбіжності з точки зору просторового подання даних про об'єкти оточуючого середовища: якщо геометричні подання повністю моделюють границі вільного простору, то подання, засновані на орієнтирах, орієнтовані на конкретні об'єкти, корисні для вирішення задач локалізації та орієнтації у просторі. Подання, засновані на орієнтирах, є доволі компактними (в залежності від щільності орієнтирів в оточуючому середовищі) і достатньо добре підходять для опису великих середовищ. Ці подання широко використовуються в підходах SLAM, оскільки дозволяють виконувати імовірнісну обробку невизначеностей. Однак використання подання, заснованого на орієнтирах, не має універсальності, властивої для подання, заснованого на зайнятості, та геометричного подання, і обмежується тільки середовищами, в яких наявні орієнтири достатньої щільності.

Реляційні подання явно визначають відношення, що існують між об'єктами оточуючого середовища. Більшість сучасних реляційних подань являють собою графові подання, в яких вузли позначають базові сутності (наприклад, види, місця, об'єкти), а ребра подають відповідні просторові відношення (наприклад, суміжність або зв'язок). Такі подання часто називають *топологічними мапами*. В таксономії (див. рис. 2) розрізняються два види подань реляційних графів: *подання графа огляду* та *подання графа маршруту*. В поданнях графа огляду вузли явно не подані в оточуючому середовищі, а утворюються в результаті його розпізнавання агентом: кожному вузлу відповідає вид, доступний для огляду з відповідної позиції

розташування агента. Виконання розподілу вузлів у вільному просторі середовища засновано на досягненні достатньої різниці між видами, отримуваними агентом з сусідніх вузлів. На відміну від цього, в поданнях графу маршруту вузли явно подані в оточуючому середовищі: граф маршруту подає середовище як мережу різних маршрутів, а вузли позначають визначні місця та особливі орієнтири, що розташовані вздовж маршруту. Часто подання графа маршруту безпосередньо відображає топологію вільного простору.

Подання графа огляду (View graph representations) характеризуються тим, що в таких графах вузли безпосередньо асоційовані з конкретною вхідною інформацією сенсора, що називається *видом* (view), отримуваною агентом з деякого місця його розташування. Зв'язок між двома вузлами враховує той факт, що обидва види були отримані в послідовному порядку і, як наслідок, наявна просторова суміжність відповідних місцеположень. Отже, зв'язок надає інформацію, необхідну для переміщення агента між цими місцезнаходженнями. Вузли мають розподілятися достатньо щільно для досягнення надійного переміщення між ними на основі навігаційних можливостей агента. Таким чином, побудова графа огляду відповідає створенню мережі вузлів, яка охоплює все оточуюче середовище. Водночас, хоча графи огляду надають інформацію для здійснення успішної навігації, отримувані при цьому шляхи руху агента, як правило, не є оптимальними. Основним недоліком подання графа огляду є те, що воно не надає структурованої інформації про оточуюче середовище.

Подання графа маршруту (Route graph representations) характеризуються тим, що в таких графах вузли позначають визначні місця, що існують у середовищі, а ребра відображають шляхи, що зв'язують такі місця, та дозволяють агенту переміщуватись з одного місця в інше. У багатьох підходах граф маршруту формується з геометрії середовища шляхом перетворення вільного простору на набір одномірних кривих, що називається *дорожньою мапою* (roadmap). Для цього, зокрема, ви-

користується діаграма Вороного та її модифікації. Подання графа маршруту дозволяє ефективно вирішувати задачі планування шляху: структура графа відповідає графу простору пошуку, на якому можна шукати шляхи за допомогою відомих методів пошуку на графах (зокрема, алгоритму Дейкстри, A^* тощо). Однією з переваг подання графа маршруту є те, що воно полегшує систематичне дослідження оточуючого середовища: для того, щоби охопити все середовище, достатньо відстежити всі ребра графа. Іншою перевагою цього подання є його компактність, що створює необхідні умови для спрощення вирішення задач комунікації агентів та обміну між ними інформацією, необхідною для побудови загальної мапи групою агентів. Водночас, проблема локалізації є достатньо складною проблемою для подання графа маршруту.

3. Огляд відомих методів формування та планування шляхів агентом

3.1. Огляд основних підходів до формування шляхів. Подання мапи, описані в п. 2 статті, використовуються агентом для вирішення задач планування шляхів в оточуючому середовищі. В більшості випадків для вирішення таких задач агент має перетворити існуючу модель оточуючого середовища (мапу) в деяке дискретне подання, придатне для здійснення планування шляхів за допомогою відомих методів пошуку. При цьому використовуються наступні підходи [6, 8]:

- *планування на основі потенційних полів* (Potential field planning). Цей підхід засновується на створенні поля або градієнту на мапі агента, що направляє агента у вільному просторі оточуючого середовища до цільової позиції з множини його попередніх позицій;

- *побудова графа пошуку* (Graph search). Цей підхід передбачає побудову графа зв'язності у вільному просторі оточуючого середовища, на якому і здійснюється пошук шляхів. Процес побудови графа пошуку часто виконується в автономному (офф-лайн) режимі.

Планування на основі потенційних полів засновується на ідеї створення штучного потенційного поля в оточуючому середовищі агента, де агент розглядається як точка, на яку впливає це поле. Ціль (глобальний мінімум у цьому полі) діє на агента як сила тяжіння, а перешкоди – як сили відштовхування. Суперпозиція всіх сил плавно направляє агента до цілі, одночасно уникаючи його зіткнень з відомими перешкодами. Треба зауважити, що використання підходу потенційних полів дозволяє забезпечити більше, ніж планування шляху. Таке поле утворює закон керування рухом агента: якщо агент може локалізувати своє місцезнаходження відносно мапи та потенційного поля, то він завжди може визначити свій наступний потрібний крок, заснований на полі. Основним недоліком підходу на основі потенційних полів є те, що у полі можуть існувати локальні мінімуми, які можуть дезорієнтувати агента у пошуку цілі.

Побудова графа пошуку лежить в основі створення дорожньої мапи та використовується в рамках кліткової декомпозиції.

При створенні *дорожньої мапи* агент з'єднує початкову та цільову позиції у вільному просторі оточуючого середовища множиною ліній, що утворюють граф зв'язності як множину шляхів, що існують між початковою та цільовою позиціями. Для створення дорожньої мапи найбільше поширення отримали підходи з побудови графа видимості (visibility graph) та діаграми Вороного (Voronoi diagram).

Граф видимості складається з множини ребер, що зв'язують всі пари вершин, що є видимими (тобто ребра проходять через вільний простір оточуючого середовища) між собою (включаючи початкову та цільові позиції як вершини). Прямі лінії (шляхи), що з'єднують ці вершини, є найкоротшими відстанями між ними і задача пошуку на утвореному графу видимості полягає у знаходженні за допомогою методів пошуку на графах найкоротшого шляху з множини шляхів, існуючих між початковою та цільовою вершинами цього графа. Завдяки досить простій процедурі його побудови, граф видимості от-

римав досить широке використання на практиці. Водночас, таке подання дорожньої мапи має свої недоліки. По-перше, розмір графу видимості зростає із збільшенням полігонів перешкод, що негативно впливає на ефективність використовуваних процедур пошуку на цьому графі. По-друге, ребра, що утворюються при побудові графу видимості, проходять на мінімальній відстані від перешкод, що негативно впливає на безпечність навігації агента по обраному найкоротшому шляху.

На відміну від графу видимості, при побудові *діаграми Вороного* забезпечується максимізація відстані між агентом та перешкодами на мапі. При побудові діаграми Вороного знаходиться множина точок, кожна з яких є центром круга, вписаного у вільний простір оточуючого середовища, який торкається як мінімум двох точок контурів перешкод. При з'єднанні послідовностей таких точок утворюються ребра, які складаються з прямих та параболічних сегментів, а вершини графу відповідають точкам зустрічі різних ребер та початковій і цільовій вершинам. На відміну від шляху, знайденого на графі видимості, найкоротший шлях, знайдений на діаграмі Вороного, не є оптимальним за критерієм довжини шляху.

Основна ідея *кліткової декомпозиції* полягає у розподілі оточуючого середовища на геометричні області (клітинки) двох категорій: вільні клітинки та зайняті клітинки. Розрізняють два види кліткової декомпозиції: точну та приблизну.

При *точній клітковій декомпозиції* створювані клітинки є або повністю вільними, або повністю зайнятими. При цьому об'єднання всіх вільних клітинок у точності дорівнює вільному простору оточуючого середовища. Для забезпечення пошуку шляхів на множині вільних клітинок будується граф суміжності, у якого ребрами є шляхи, що існують між суміжними клітинками, а вершинами графу є такі клітинки. Пошук шляху на графі полягає у знаходженні послідовності суміжних клітинок, через які проходить шлях з початкової позиції у цільову. Основним недоліком точної кліткової декомпозиції є те, що кількість клітинок і, як наслідок, загальна ефе-

ктивність планування шляхів залежить від щільності та складності об'єктів оточуючого середовища.

При *приблизній клітковій декомпозиції*, на відміну від точної кліткової декомпозиції, клітинки мають наперед визначену форму (наприклад, форму прямокутника). Якщо в процесі розбиття з'ясовується, що отримана клітинка є повністю вільною, то вона далі не декомпозується. В супротивному випадку виконується подальша декомпозиція такої клітинки на менші клітинки такої ж форми. Побудова графу зв'язності, який проходить тільки через повністю вільні клітинки, виконується за аналогією з побудовою графу суміжності у випадку точної кліткової декомпозиції. Пошук шляху на такому графі завершується успішно, якщо існує шлях між початковою та цільовою позиціями. Неуспішний результат може означати або відсутність такого шляху, або недостатність виконаної декомпозиції. Таким чином, в рамках приблизної кліткової декомпозиції пошук шляхів може виконуватись ієрархічно шляхом поступової деталізації ступені декомпозиції. В наслідок цього приблизна кліткова декомпозиція є однією з найбільш популярних технік з побудови графу пошуку.

3.2. Огляд найбільш відомих методів планування шляхів. Для пошуку шляхів на графах, створюваних за допомогою підходів, описаних в п. 3.1, використовуються різні методи (алгоритми), в тому числі [8 – 10]: пошуку в ширину, пошуку в глибину, Дейкстри, A^* , D^* .

Метод пошуку в ширину лежить в основі алгоритму обходу графу, який починає свою роботу з кореневої вершини графу, «розкриваючи» її, та відвідує всі її суміжні вершини, послідовно «розкриваючи» кожен з них і так далі. Таким чином, при пошуку в ширину перед «розкриттям» будь-яких вершин наступного рівня відбувається «розкриття» всіх вершин попереднього рівня графу. Алгоритм пошуку завершує свою роботу тоді, коли буде знайдена цільова вершина. Зазначимо, що алгоритм пошуку в ширину завжди формує шлях з мінімальною кількістю ребер між кореневою та цільовою вершинами. Недо-

ліком методу пошуку в ширину є велика потреба у пам'яті, необхідної для збереження оброблених даних: цей метод потребує збереження всіх відвіданих вершин.

Метод пошуку в глибину лежить в основі алгоритму обходу графа, який починає свою роботу з деякої кореневої вершини (таких вершин може бути декілька) і намагається йти «в глиб» графа наскільки це можливо. При виконанні алгоритму досліджуються всі ребра, що виходять з i -ої вершини, «розкритої» останньою, і залишає вершину тільки тоді, коли не залишилось недосліджених ребер, при цьому відбувається повернення до вершини, з якої була «розкрита» i -та вершина. Процес обходу графа триватиме до тих пір, доки не будуть «розкриті» всі вершини, досяжні з обробленої кореневої вершини. Якщо при цьому залишаються «нерозкриті» вершини (наприклад, інші кореневі вершини), то одна з них обирається як нова початкова вершина і процес обходу графа повторюється вже відносно неї. Алгоритм пошуку, як і у випадку пошуку в ширину, завершує свою роботу тоді, коли буде знайдена цільова вершина. Недоліком методу пошуку в глибину є те, що він може повторно «розкрити» раніше відвідані вершини та створити зайві шляхи, хоча ці недоліки можуть бути легко усунені при програмній реалізації алгоритму. Перевагою цього методу є невелика потреба у пам'яті, необхідної для збереження оброблених даних: цей метод потребує збереження тільки єдиного шляху від кореневої вершини до термінальної наряду з усіма «нерозкритими» суміжними вершинами для кожної вершини цього шляху.

Алгоритм Дейкстри подібний за принципом обходу графа методу пошуку в ширину за винятком того, що він здійснює пошук найкоротшого шляху з початкової до цільової вершини на зваженому графі, в якому ваги ребер невід'ємні. Алгоритм Дейкстри знаходить найкоротші шляхи до вершин графа в порядку їх віддалення від початкової вершини. На кожному кроці ітерації алгоритм визначає найкоротші шляхи з можливих альтернатив завдяки знаходженню шляхів з мінімальною сумарною вагою ребер. В загальному випадку

перед початком i -ої ітерації алгоритм визначає найкоротші шляхи до $i-1$ інших вершин, найближчих до початкової. Ці вершини, початкова вершина та ребра найкоротших шляхів, що ведуть до цих вершин з початкової вершини, утворюють підграф даного графа. Вершина, найближча до початкової, може бути знайдена серед вершин, суміжних кінцевій вершині цього підграфа. Для визначення i -ої найближчої вершини алгоритм обчислює для кожної суміжної вершини суму ваги ребра, що з'єднує таку вершину з найближчою вершиною підграфа, та мінімальної сумарної ваги ребер (раніше визначеної алгоритмом), що ведуть від початкової вершини до цієї найближчої вершини підграфа, і далі обирає вершину з мінімальною сумою. Алгоритм завершує свою роботу тоді, коли будуть досліджені всі шляхи, що ведуть від початкової до цільової вершини, та визначено найкоротший шлях як шлях з мінімальною сумарною вагою утворюючих його ребер.

*Алгоритм A^** є удосконаленням алгоритму Дейкстри за рахунок введення евристичної функції, що враховує властивості графа пошуку і дозволяє зменшити кількість досліджуваних вершин графа. В алгоритмі A^* порядок обходу вершин графа визначається евристичною функцією $f(n)$ як сумою функції $g(n)$ вартості шляху від початкової до даної вершини n та функції $h(n)$ як евристичної оцінки вартості шляху від цієї вершини n до цільової: $f(n) = g(n) + h(n)$. Алгоритм A^* досліджує граф ітеративно, розглядаючи шляхи, що виходять з початкової вершини. На кожному кроці ітерації алгоритм розглядає вершини, суміжні поточній вершині, та обирає ту з них, для якої $f(n)$ є мінімальною, після чого ця вершина «розкривається». В загальному випадку на кожному кроці ітерації алгоритм оперує з множиною шляхів від початкової вершини до ще не «розкритих» вершин графа, такі шляхи зберігаються в черзі з пріоритетом, де пріоритет визначається за значенням $f(n)$. Алгоритм продовжує свою роботу до тих пір, доки не досягне цільової вершини і не визначить шлях до неї з найменшим значен-

ням $f(n)$. Недоліком алгоритму A^* є те, що евристичні оцінки вартості шляхів задані остаточно і не можуть бути змінені в процесі роботи алгоритму (без його перезапуску), що не дозволяє коректно планувати шляхи в середовищі, яке динамічно змінюється.

Алгоритм D^* за назвою походить від терміну «динамічний A^* », оскільки цей алгоритм подібний алгоритму A^* за винятком того, що ваги ребер можуть змінюватись в процесі виконання алгоритму. На відміну від алгоритму A^* , в алгоритмі D^* пошук шляхів виконується від цільової вершини. При цьому на зваженому графі шукається шлях мінімальної вартості. Якщо в процесі руху агента по такому шляху виявляються перешкоди, то алгоритм змінює ваги відповідних ребер і динамічно перевизначає оптимальний шлях, враховуючи нову інформацію. Процес повторюється до тих пір, доки агент не досягне цільової вершини, або з'ясує, що це неможливо.

4. Постановка задач досліджень

На даному етапі досліджень ми розглядаємо проблему SMAP (див. вступ) для випадку одного агента. При цьому ми зауважуємо, що технічні аспекти вирішення цієї проблеми виходять за рамки даних досліджень, тобто вважаємо, що вирішення задач точної локалізації, одометрії, усунення похибок при визначенні кутів поворотів та відстаней до об'єктів оточуючого середовища, стосуються якості функціонування GPS-обладнання, одометрів і сенсорів та вирішуються на рівні апаратного забезпечення досліджуваних процесів. Нашими задачами є теоретичні аспекти вирішення проблеми SMAP, які на даному етапі досліджень полягають у розробці методів і моделей коректного вирішення задач одночасної побудови мапи невідомого оточуючого середовища та планування шляху в ньому для випадку одного агента, на якість дій якого не впливають вищезгадані похибки вимірювання, тобто який в кожний момент часу точно знає своє місцезнаходження та спроможний використовувати власні алгоритми для коректного розпізнавання невідомого оточуючого середовища та навігації у ньому.

Для постановки задач досліджень будемо виходити з наступних припущень:

- оточуючим середовищем для агента є будівлі ортогональної конфігурації в 2D проекції з довільною кількістю стін та отворів;
- оточуюче середовище і агент знаходяться в єдиній Декартовій системі координат.

Будемо вважати, що завжди існує мапа M оточуючого середовища як фізичного об'єкта, яка невідома агенту і яку він має розпізнати, на основі чого побудувати власну мапу цього середовища. Мапа M описує множину довільних фрагментів стін, які фактично існують у межах оточуючого середовища. Підкреслимо, що задання мапи як множини фрагментів стін не є випадковим, – таким чином ми намагаємось формалізувати процес сприйняття досліджуваного оточуючого середовища людиною: внаслідок властивостей зору, людина не може, наприклад, одномоментно сприйняти всю стіну цілком, а сприймає її як послідовність фрагментів. Виходячи з цього,

$$M = \{w_1, w_2, \dots, w_n\},$$

де $w_i = \langle id_i, (x_1^i, y_1^i), (x_2^i, y_2^i) \rangle$, $w_i \in M$, id_i – унікальний ідентифікатор w_i .

Задача r розпізнавання невідомого оточуючого середовища та навігації у ньому може бути описана як

$$M \xrightarrow{r} \langle M_a, T_a, G_a \rangle,$$

де M_a – мапа оточуючого середовища, побудована агентом a в результаті розпізнавання цього середовища та навігації у ньому; T_a – траєкторія руху агента a , сформована ним у процесі вирішення задач розпізнавання оточуючого середовища та навігації у ньому; G_a – граф простору пошуку шляхів, сформований агентом a у процесі розпізнавання оточуючого середовища та навігації у ньому.

Мапа

$$M_a = \{wr_1, wr_2, \dots, wr_m\},$$

де $wr_j = \langle idr_j, (x_1^j, y_1^j), (x_2^j, y_2^j) \rangle$, $wr_j \in M_a$, idr_j – унікальний ідентифікатор wr_j . Мапа M_a включає множину розпізнаних стін wr_j , кінці (x_1^j, y_1^j) , (x_2^j, y_2^j) яких, на відміну від фрагментів стін, що належать мапі M , обов'язково утворюють деякі кути приміщення, що розпізнається. Будемо умовно розрізняти внутрішні та зовнішні кути приміщення (див. рис. 3). Внутрішні кути відповідають так званим «глухим» кутам у приміщенні, головною ознакою яких є те, що вони унеможливають подальший рух агента в їх напрямку. На відміну від цих кутів, головна ознака зовнішніх кутів – те, що кожен з них є кутом деякого отвору, через який агент може перейти в інше приміщення або в іншу частину даного приміщення; виявлення таких кутів важливо для вирішення задач планування агентом своїх подальших шляхів (формування графа G_a).

На рис. 3 показано приклад виявлення внутрішніх та зовнішніх кутів у довільному приміщенні, та умовно показано отвори (A – B, C – D, E – F, G – H), що відповідають виявленим зовнішнім кутам (або їх парним точкам).

Для розпізнавання оточуючого середовища та побудови мапи M_a агент у кожний момент часу свого руху направляє промені в оточуюче середовище (вважатимемо, що таким чином ми моделюємо

«технічний зір» агента, що на практиці реалізується за допомогою відповідних сенсорів), причому крок цих променів має динамічно змінюватись у процесі розпізнавання. Дійсно, якщо крок променів прийняти постійним, то можуть виникати ситуації «загублення» інформації.

На рис. 4, а показано випадок, коли агент розташований у різних точках A та B отримує на вхід («бачить») різну інформацію про оточуюче середовище, що відповідає ситуації, коли крок променя постійний, в результаті чого агент у позиції B не «бачить» отвору. На рис. 4, б показано випадок динамічної зміни кроку променя, в результаті чого незалежно від місця розташування агент отримує на вхід однакову інформацію про оточуюче середовище. Таким чином ми намагаємось врахувати властивості зору людини, пов'язані з *акомодацією* очей при розгляді об'єктів на різних відстанях (при цьому відбувається зміна фокусної відстані ока).

Неформально процес розпізнавання оточуючого середовища та побудови мапи агентом полягає у тому, що на кожному кроці свого руху у вільному просторі оточуючого середовища агент з певним змінним кроком направляє промені відносно себе та визначає відстані до найближчих об'єктів (перешкод) оточуючого середовища, що належать мапі M , та координати точок перетинів променя та перешкоди.

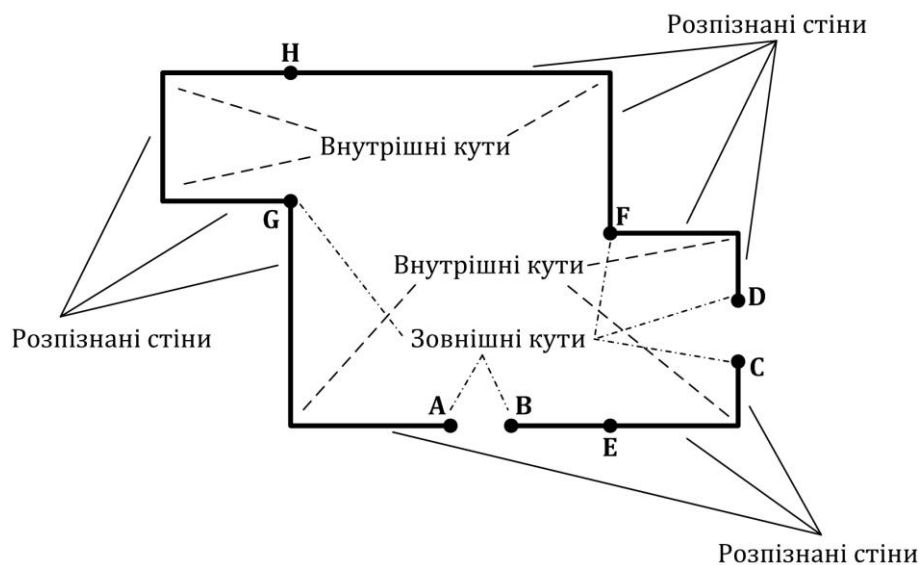


Рис. 3. Приклади кутів у довільному приміщенні

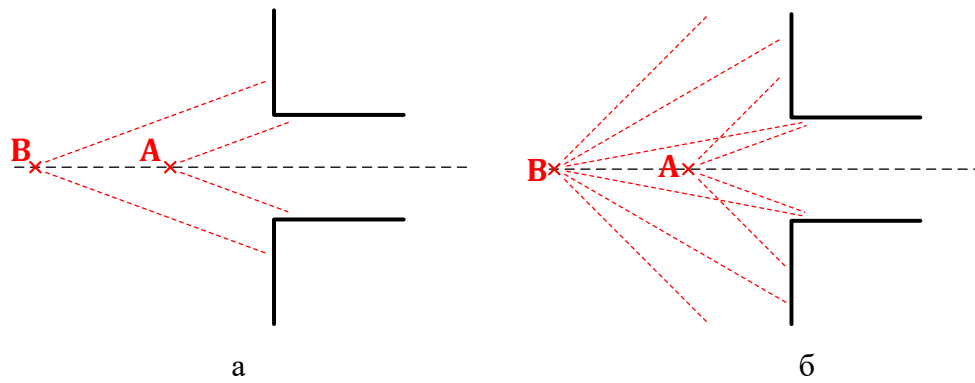


Рис. 4. Приклади різних розташувань агента та напрямів променів

Внаслідок того, що промені направляються послідовно, агент узагальнює координати перетинів, отримані від попередніх променів та поточного променя, та формує відповідні елементи оточуючого середовища як компоненти мапи M_a . В процесі формування мапи агент аналізує сформовані компоненти на можливість утворення ними отворів. Якщо отвір виявлено, агент додає до графа G_a новий фрагмент шляху у складі двох його початкових елементів.

Очевидно, що запропоноване подання мапи можна розглядати як комбінацію геометричного подання та подання, заснованого на орієнтирах (див. п. 2 статті).

Траскторія T_a руху агента містить множину послідовних за часом позицій агента, сформованих ним у процесі руху в середовищі:

$$T_a = \{pos_1, pos_2, \dots, pos_s\},$$

де $pos_t = \langle (x_t, y_t), \Theta_t, v_t \rangle$, $pos_t \in T_a$. Тут (x_t, y_t) – координати розташування агента a в момент часу t ; Θ_t – кут напрямку руху агента a в момент часу t ; v_t – швидкість руху агента a в момент часу t . Зазначимо, що оскільки як оточуюче середовище розглядаються довільні будівлі *ортогональної* конфігурації, то для моделювання напрямку руху (кута Θ_t) агента будемо розглядати тільки 4 варіанти: «праворуч» (кут руху 0°), «уверх» (кут руху 90°), «ліворуч» (кут руху 180°), «униз» (кут руху 270°). Для вирішення проблеми локалізації будемо виходити з вимог, що відповідають

локальній локалізації (див. п. 2 статті), тобто агент на початку свого руху (в момент часу t_0) «знає» своє місцезнаходження (координати (x_{t_0}, y_{t_0})), а в подальшому визначає його алгоритмічно. Будемо вважати, що швидкість v_t руху агента a може бути змінною (у тому числі й нульовою), оскільки в загальному випадку він може, наприклад, зупинитися при зустрічі з іншим динамічним об'єктом.

Граф G_a простору пошуку шляхів може бути описаний як кінцева множина упорядкованих фрагментів шляхів, сформованих агентом у процесі розпізнавання невідомого оточуючого середовища та навігації у ньому:

$$G_a = \{Path_1, Path_2, \dots, Path_q\},$$

де $Path_k \in G_a$ – упорядкована послідовність координат позицій агента, займаних ним у процесі розпізнавання окремої частини невідомого оточуючого середовища. Кожний $Path_k \in G_a$ виступає як ребро графа G_a ; максимально можлива кількість ребер q у такому графі дорівнює $2Z - 1$, де Z – загальна кількість отворів, що належать до оточуючого середовища, яке розпізнається. Зазначимо, що процес формування графа простору пошуку шляхів доцільно виконувати, враховуючи використовуваний принцип, за яким на такому графі буде виконуватись пошук.

В нашому випадку як метод пошуку на графі обрано пошук у глибину, оскільки, як показано в п. 3.2, він має переваги з точки зору використовуваної

пам'яті, необхідної для обробки графа. Нагадаємо, що такий граф має будуватися та оброблятися за принципом LIFO.

Для ілюстрації процесу формування графа G_a розглянемо приклад на рис. 5. На рисунку «хрестики» позначають позиції, відвідані агентом, «кружки» – позиції, заплановані агентом для відвідання в процесі подальшого обходу графа.

Припустимо, що агент розпізнав Отвір 1 та рухається під кутом 90° в Декартовій системі координат і в процесі подальшого розпізнавання середовища формує траєкторію руху, послідовно займаючи координати позицій $(x_1, y_1), (x_1, y_2), \dots, (x_1, y_b)$ та формуючи при цьому фрагмент шляху $Path_{1_1}$, який включається до складу графа G_a . В момент часу, коли агент досяг координати (x_1, y_b) , він розпізнає Отвір 2, і за принципом LIFO додає до графа G_a фрагмент нового шляху $Path_2$ у складі координат його майбутніх запланованих позицій

$\{(x_2, y_b), (x_1, y_b)\}$, тобто на цей момент часу граф простору пошуку шляхів прийме вигляд $G_a = \{Path_2, Path_{1_1}\}$. Продовжуючи подальший рух, агент формує шлях $Path_{1_2} = \{(x_1, y_c), \dots, (x_1, y_b)\}$, який включається до складу графу G_a за принципом LIFO, в деякий момент часу досягаючи координати (x_1, y_c) та розпізнаючи Отвір 3, у результаті чого додає до графа G_a фрагмент нового шляху $Path_3$ у складі координат його майбутніх запланованих позицій $\{(x_3, y_c), (x_1, y_c)\}$. При цьому послідовність шляхів, що формують граф G_a , прийме наступний вигляд:

$$G_a = \{Path_3, Path_{1_2}, Path_2, Path_{1_1}\}.$$

Подальший рух агента призведе до поступового додавання відповідних координат до складу шляхів, а шляхів – до складу графа G_a , і в позиції агента (x_1, y_g) остаточний склад графа G_a , сформований за

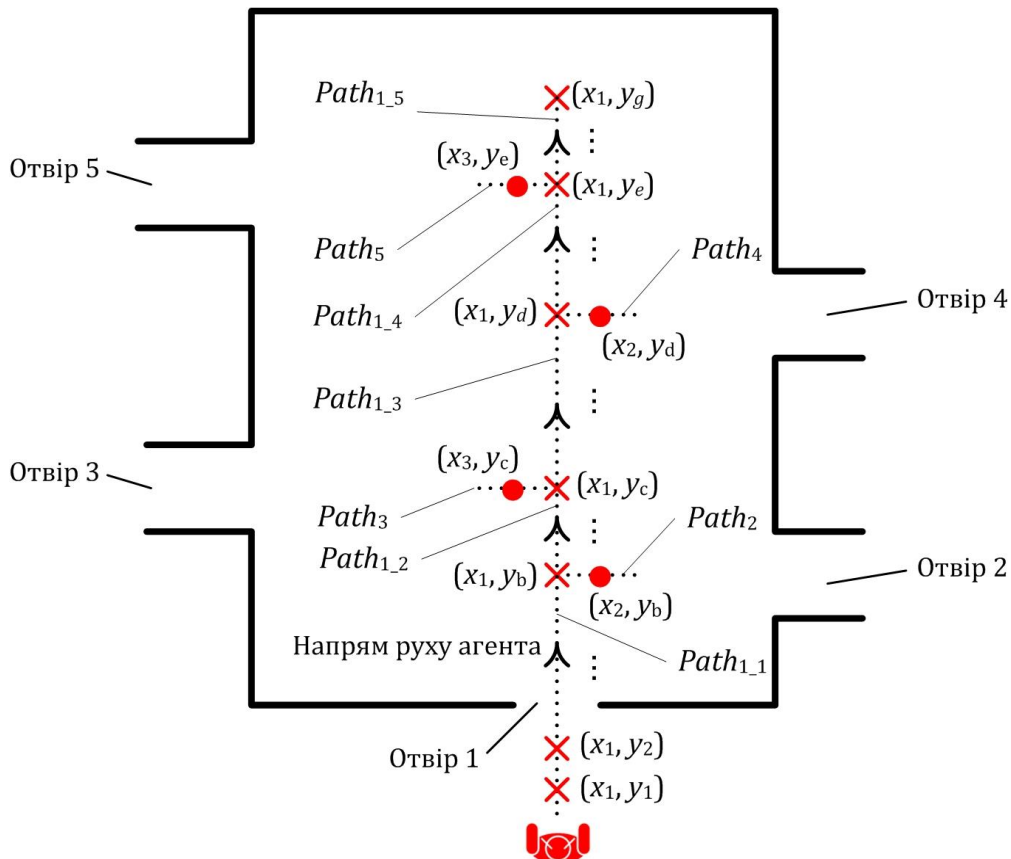


Рис. 5. Приклад формування графу пошуку шляхів

результатами планування шляхів у досліджуваному приміщенні, прийме наступний вигляд:

$$G_a = \{Path_{1_5}, Path_5, Path_{1_4}, Path_4, Path_{1_3}, Path_3, Path_{1_2}, Path_2, Path_{1_1}\}.$$

Виходячи з даного прикладу, легко помітити, що побудована послідовність фрагментів шляхів відповідає графу простору пошуку шляхів, в якому кожне ребро відповідає окремому фрагменту шляху, а внутрішні вершини – загальним елементам цих шляхів. Також легко помітити, що подальший пошук на сформованому графі може бути виконаний за допомогою методу пошуку в глибину.

Таким чином, виходячи з вищевикладеного, можна сформулювати наступний перелік методів, які необхідно розробити на даному етапі досліджень:

- метод визначення агентом відстані до найближчого від нього об'єкта та координат перетину з цим об'єктом променя, направленою від агента;
- метод динамічної зміни агентом кроку променя, який ним направляється в оточуюче середовище;
- методи розпізнавання агентом кутів приміщень в оточуючому середовищі;
- методи узагальнення агентом кутів приміщень з виявленням отворів у оточуючому середовищі;
- методи узагальнення агентом інформації про оточуюче середовище з побудовою мапи цього середовища;
- методи формування агентом траєкторії руху, що ґрунтуються на врахуванні ним узагальненої інформації про оточуюче середовище;
- методи планування агентом шляхів пошуку, що ґрунтуються на врахуванні ним узагальненої інформації про оточуюче середовище.

Висновки

У статті виконано аналітичний огляд проблеми розпізнавання невідомого оточуючого середовища, навігації та пла-

нування шляхів агентом у ньому та визначено місце наших досліджень в рамках вирішення цієї проблеми. Обґрунтовано, що обраний нами напрям досліджень належить до проблеми *одночасної побудови мапи та планування шляхів* (simultaneous mapping and planning (SMAP)). Виходячи із структури досліджуваної проблеми, розглянуто сутність проблеми локалізації та визначено підхід до вирішення цієї проблеми в рамках виконуваних досліджень. Виконано огляд відомих підходів до побудови та подання мапи оточуючого середовища та розглянуто основні сучасні підходи до формування шляхів пошуку на мапі й основні методи планування шляхів. На основі цих оглядів запропоновано нову постановку задач досліджень, в якій для подання мапи використовується підхід, який може розглядатися як комбінація геометричного подання та подання простору пошуку використовується графове подання, що за правилами побудови відрізняється від інших відомих підходів з формування графа пошуку. Як впливає з запропонованої постановки задачі, вона належить до проблематики SMAP, оскільки виходячи з неї агент одночасно здійснює побудову мапи оточуючого середовища та планування шляхів.

Література

1. Stachniss C. *Robotic Mapping and Exploration*. Springer. 2009. 196 p.
2. Makarenko A.A., Williams S.B., Bourgault F., Durrant-Whyte H.F. An Experiment in Integrated Exploration. *Proceedings of the 2002 IEEE/RSJ International Conference on Intelligent Robots and Systems*. EPFL. Switzerland, 2002. P. 534–539.
3. Durrant-Whyte H., Bailey T. Simultaneous Localization and Mapping: Part I. *IEEE Robotics & Automation Magazine*. 2006. Vol. 13, Iss. 2. P. 99–108.
4. Bailey T., Durrant-Whyte H. Simultaneous Localization and Mapping: Part II. *IEEE Robotics & Automation Magazine*. 2006. Vol. 13, Iss. 3. P. 108–117.
5. Thrun S., Burgard W., Fox D. *Probabilistic*

- Robotics. MIT Press. 2006. 647 p.
6. Siegwart R., Nourbakhsh I.R., Scaramuzza D. Introduction to Autonomous Mobile Robots. MIT Press. 2011. 453 p.
 7. Wallgrün J.O. Hierarchical Voronoi Graphs: Spatial Representation and Reasoning for Mobile Robots. Springer, 2010. 218 p.
 8. Latombe J.C. Robot motion planning. Springer Science + Business Media, LLC. 1991. 651 p.
 9. Кормен Т., Лейзерсон Ч., Ривест Р., Штайн К. Алгоритмы: построение и анализ. М.: Вильямс, 2013. 1329 с.
 10. LaValle S.M. Planning Algorithms. Cambridge University Press. 2006. 826 p.

- Mobile Robots. Springer, 2010. 218 p.
8. Latombe J.C. Robot motion planning. Springer Science + Business Media, LLC. 1991. 651 p.
 9. Cormen T., Leiserson Ch., Rivest R., Stein C. Introduction to Algorithms. M. Williams. 2013. 1329 p. (in Russian)
 10. LaValle S.M. Planning Algorithms. Cambridge University Press. 2006. 826 p.

Одержано 10.01.2018

References

1. Stachniss C. Robotic Mapping and Exploration. Springer. 2009. 196 p.
2. An Experiment in Integrated Exploration / Makarenko A.A., Williams S.B., Bourgault F., Durrant-Whyte H.F. // Proceedings of the 2002 IEEE/RSJ International Conference on Intelligent Robots and Systems. EPFL. Switzerland, 2002. P. 534–539.
3. Durrant-Whyte H., Bailey T. Simultaneous Localization and Mapping: Part I // IEEE Robotics & Automation Magazine. Vol.13. Iss. 2. 2006. P. 99–108.
4. Bailey T., Durrant-Whyte H. Simultaneous Localization and Mapping: Part II // IEEE Robotics & Automation Magazine. Vol. 13. Iss. 3. 2006. P. 108–117.
5. Thrun S., Burgard W., Fox D. Probabilistic Robotics. MIT Press. 2006. 647 p.
6. Siegwart R., Nourbakhsh I.R., Scaramuzza D. Introduction to Autonomous Mobile Robots. MIT Press. 2011. 453 p.
7. Wallgrün J.O. Hierarchical Voronoi Graphs: Spatial Representation and Reasoning for

Про автора:

Яловець Андрій Леонідович,
доктор технічних наук,
заступник директора інституту.
Кількість наукових публікацій в
українських виданнях – понад 100.
Кількість наукових публікацій в
зарубіжних виданнях – 10.
<http://orcid.org/0000-0001-6542-3483>.

Місце роботи автора:

Інститут програмних систем
НАН України.
03187, Київ-187,
проспект Академіка Глушкова, 40.

Тел.: (044) 526 15 38.
E-mail: yal@isofts.kiev.ua