

ПЛАНУВАННЯ ПОСЛІДОВНОСТІ ВИКОНАННЯ ЗАВДАНЬ ІНТЕЛЕКТУАЛЬНИМ АГЕНТОМ З УРАХУВАННЯМ КОНТЕКСТУ

О. Ю. Бочкарьов

Національний університет “Львівська політехніка”,
кафедра електронних обчислювальних машин
E-mail: oleksii.y.bochkarov@lpnu.ua

© Бочкарьов О. Ю., 2022

Розглянуто проблему контекстно-залежного планування послідовності виконання незалежних або слабопов'язаних завдань інтелектуальним агентом. Проаналізовано принципи відповідності завдання контексту. Запропоновано структуру модуля контекстно-залежного планування послідовності виконання завдань у складі інтелектуального агента та алгоритм його роботи. У структурі модуля планування реалізовано три основні групи блоків: робота з завданнями, робота з контекстом, визначення відповідності завдань контексту.

Також в статті запропоновано алгоритм розрахунку динамічного пріоритету завдання, алгоритм визначення відповідності завдання контексту і алгоритм адаптації набору правил відповідності завдань контексту. Значення динамічного пріоритету залежить від статичного пріоритету завдання, що призначається в момент додавання завдання, та ступеня відповідності завдання контексту з урахуванням вектору контексту. Допускається два режими запуску алгоритму планування: за фактом звернення інтелектуального агента за новим завданням та за фактом зміни вектору контексту. Алгоритм розрахунку динамічного пріоритету виконується незалежно для кожного завдання. Внаслідок цього його програмна реалізація має великий ресурс розпаралелювання.

Для адаптації набору правил відповідності завдань контексту використана схема навчання з підкріпленням в стаціонарному випадковому середовищі з контекстною залежністю (contextual multi-armed bandit problem). Для кожного правила відповідності виконується свій примірник процедури навчання з підкріпленням. В якості методу навчання з підкріпленням в статті використано метод верхньої довірчої межі (Upper-Confidence-Bound Action Selection). Запропоновано схему роботи процедури навчання з підкріпленням, реалізовану в прототипі модуля контекстно-залежного планування послідовності виконання завдань інтелектуальним агентом. Схема роботи процедури навчання з підкріпленням дозволяє використовувати декомпозицію даних і функціональну декомпозицію для розпаралелювання відповідних обчислень.

Ключові слова: контекстна залежність, інтелектуальний агент, планування послідовності виконання завдань.

Вступ

Один з основних напрямків розвитку ідей і технологій штучного інтелекту полягає у створенні автономних інтелектуальних агентів, здатних самостійно без участі людини вирішувати поставлені перед ними завдання [1-3]. Прикладами інтелектуальних агентів можуть бути автономні роботи, безпілотні літальні апарати, автономні підводні апарати, програмні агенти, персональні помічники та ін. Сучасні досягнення в області комп'ютерних технологій, технологій бездротового зв'язку, мобільних обчислень та в інших областях дають можливість реалізувати все більш

складні архітектури інтелектуальних агентів і нарощувати їх функціональність. В результаті часто виникають ситуації, коли перед інтелектуальним агентом стоїть одночасно кілька різних завдань, і йому потрібно вибрати деяку послідовність їх виконання. Однією з перших областей, у якій дослідники зіткнулися з цією проблемою, була розробка автономних роботів для складального виробництва [4-7]. При цьому в залежності від специфіки завдань і характеру зв'язків між ними можуть розглядатися різні задачі оптимізації їх послідовності. Серед них особливо актуальною є задача планування послідовності виконання завдань з урахуванням стану оточення агента та спостережуваної динаміки його змін. В даному випадку мається на увазі, що агент здатний визначати яке із завдань в даних умовах найбільш вигідно виконати раніше за інших, і формувати черговість виконання завдань відповідним чином. Іншими словами планування в даному випадку є контекстно-залежним і може бути предметом дослідження в рамках більш загальної концепції контекстно-залежних обчислень [8-13].

В статті розглянута проблема контекстно-залежного планування послідовності виконання незалежних або слабопов'язаних завдань інтелектуальним агентом. В тому числі проаналізовано принцип відповідності завдання контексту, запропонована структура модуля планування послідовності виконання завдань у складі інтелектуального агента та алгоритм його роботи. В статті запропоновано алгоритм розрахунку динамічного пріоритету завдання, алгоритм визначення відповідності завдання контексту і алгоритм адаптації набору правил відповідності завдань контексту на основі навчання з підкріпленням в стаціонарному випадковому середовищі з контекстно залежністю (contextual multi-armed bandit problem). Запропоновано схему роботи процедури навчання з підкріпленням, що реалізована в прототипі модуля планування послідовності виконання завдань. Слід зазначити, що отримані результати можна узагальнити на випадок планування послідовності виконання завдань багатоагентною системою, в тому числі з урахуванням розподілу окремих завдань між агентами.

1. Проблема контекстно-залежного планування

Розглянемо множину незалежних завдань $Q=\{q\}_n$, які потрібно виконати інтелектуальному агенту. Будемо вважати, що кожне завдання q характеризується непустим набором атрибутів з множини всіх атрибутів $X=\{x\}_n$. Прикладами атрибутів можуть бути тип ресурсу необхідний для виконання завдання, оцінка витрат часу на виконання завдання, необхідність взаємодії з користувачем для виконання завдання та ін. Виходячи з концепції контекстно-залежних обчислень, розглянемо модель контексту у вигляді множини параметрів контексту $C=\{c\}_m$, кожен з яких може приймати одне з декількох заданих значень. Прикладами параметрів контексту можуть бути показники апаратних або програмних сенсорів, інформація про місцезнаходження інтелектуального агента в просторі, поточний час, інформація про наявність доступу до ресурсу та ін. При цьому в залежності від області застосування інтелектуального агента можуть бути використані різні за структурою і змістом моделі контексту [8,10,12]. З урахуванням того, що в кожен момент часу всі параметри $\{c\}_m$ приймають якісь значення, будемо розглядати простір контексту (context space) (S,d) , де $S=\{s\}$ – множина векторів контексту (context vectors), а d – метрика на цій множині (в загальному випадку евклідова), така що $d(s_i,s_j)=\|s_i-s_j\|$. У випадку багатоагентної системи [3] можна додатково розглядати спільний контекст кількох агентів, для визначення якого необхідно забезпечити можливість обміну інформацією між ними.

В основу контекстно-залежного планування покладено принцип відповідності завдання контексту. Мається на увазі, що між атрибутами завдань і значеннями параметрів контексту є певний зв'язок, який дозволяє розрізняти завдання з точки зору їх більшої чи меншої відповідності тим умовам, в яких в даний момент знаходиться інтелектуальний агент. Цей зв'язок зручно представити у вигляді множини правил відповідності завдань контексту $U=\{u\}_k$. Кожне правило u описує

окремий аспект відповідності завдань контексту у вигляді комбінації значень параметрів контексту з S та відповідної комбінації атрибутів завдань з X . Чим більше правил вказують на відповідність завдання q вектору контексту $s(t)$, тим вигідніше в даних умовах виконати це завдання раніше за інших. У базовому сценарії розробник задає набір правил U агенту перед початком його роботи. При цьому вигідно реалізувати можливість подальшої адаптації набору правил U під специфіку виконуваних агентом завдань в попередньо невідомих розробнику умовах роботи агента. Адаптований набір правил відповідності завдань контексту на кроці t позначимо як $U_t \subseteq U$. Додатково можна розглянути цікаву можливість самостійного формування правил $\{u\}$ агентом і їх перевірки в процесі виконання завдань (автономне дослідження відповідності завдань контексту).

Як результат планування послідовності виконання завдань, будемо розглядати впорядкований список (множину) актуальних завдань $Q_A = \{q\}_z, 1 \leq z \leq n, Q_A \subseteq Q$. Чим вище завдання у списку Q_A , тим доцільніше або вигідніше його виконати в даних умовах (поточній ситуації, контексті). Таким чином, проблема контекстно-залежного планування полягає у формуванні списку актуальних завдань Q_A з множини Q згідно адаптованому набору правил U_t відповідно до вектору контексту $s(t)$.

2. Модуль планування послідовності виконання завдань

Для вирішення сформульованої проблеми вигідно виділити в структурі інтелектуального агента окремий модуль контекстно-залежного планування. У структурі самого модуля планування (рис. 1) можна виділити три основні групи блоків:

- 1) робота з завданнями, в тому числі додавання/видалення завдань, формування нових завдань, розрахунок динамічних пріоритетів завдань і, власне, планування послідовності їх виконання;
- 2) робота з контекстом, в тому числі визначення значень параметрів контексту за допомогою сенсорних підсистем, а також шляхом об'єднання інформації, що надходить від різних сенсорів (sensor fusion), і формування вектору контексту $s(t)$, виходячи із заданої моделі контексту S ;
- 3) визначення відповідності завдань контексту, в тому числі додавання/видалення правил відповідності, і формування адаптованого набору правил U_t шляхом оцінки ефективності роботи інтелектуального агента для заданого списку актуальних завдань і виконання процедури навчання з підкріпленням.

3. Планування послідовності виконання завдань

Для контекстно-залежного планування послідовності виконання завдань можна запропонувати наступний алгоритм (рис. 2). Ідея алгоритму полягає в тому, що для кожного завдання q розраховується його динамічний пріоритет $P_d(q)$, $P_d(q) \in [0,1]$, після чого завдання сортуються за спаданням пріоритету, і z перших з них потрапляють в список актуальних завдань Q_A . При цьому значення динамічного пріоритету залежить від двох величин: 1) статичного пріоритету завдання $P_s(q) = \text{const}$, $P_s(q) \in [0,1]$, що призначається в момент додавання завдання q в множину Q , 2) ступеня відповідності завдання контексту $R(q) \geq 0$ з урахуванням вектору контексту $s(t)$. При цьому допускається два режими запуску алгоритму: 1) за фактом звернення інтелектуального агента за новим завданням, 2) за фактом зміни вектору контексту. Також з точки зору формування списку актуальних завдань Q_A можна запропонувати два варіанти дій: 1) в список Q_A додаються перші z завдань з найбільшим динамічним пріоритетом; 2) в список Q_A додаються завдання, динамічний пріоритет яких більше заданого порогового значення. Варіанти використання агентом списку актуальних завдань Q_A : 1) агент вибирає завдання для виконання з Q_A з ймовірністю $1/z$, 2) агент вибирає з Q_A завдання з найбільшим динамічним пріоритетом (перше в списку), 3) комбінація першого і другого варіанту.

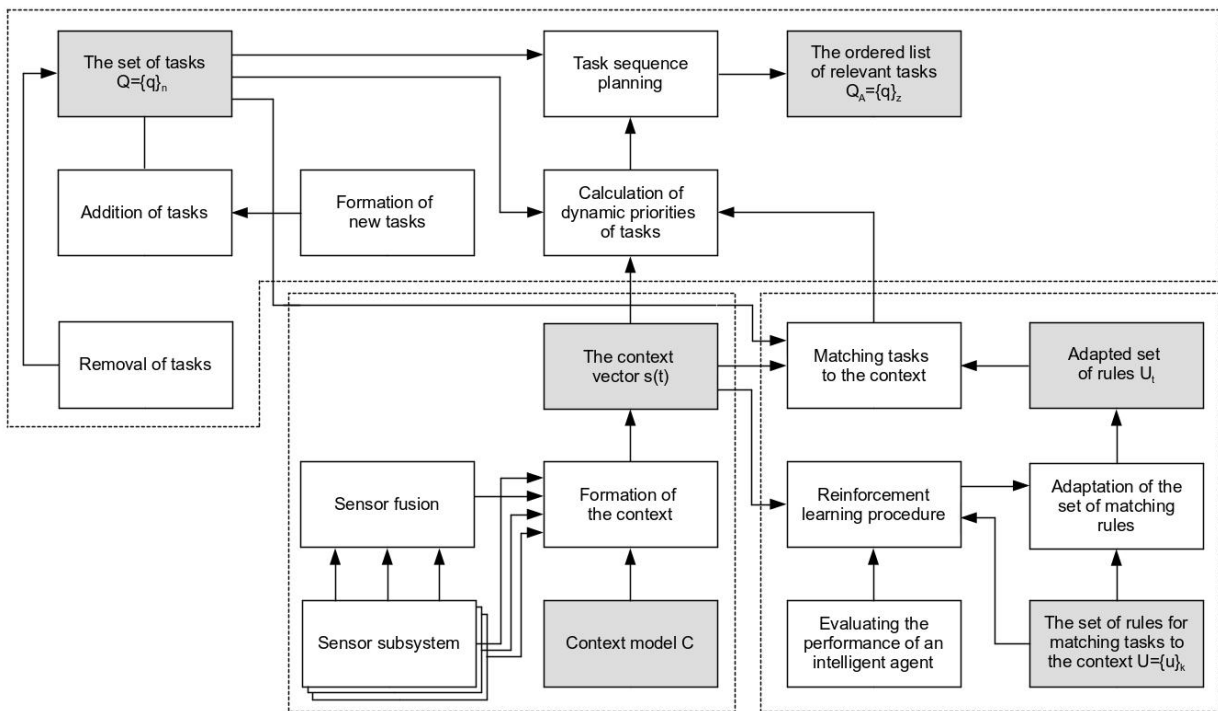


Рис. 1. Структура модуля контекстно-залежного планування послідовності виконання завдань

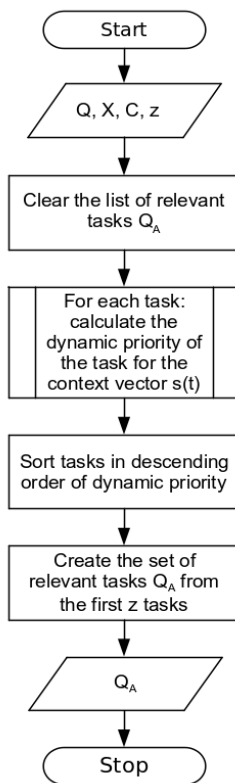


Рис. 2. Алгоритм контекстно-залежного планування послідовності виконання завдань (A1)

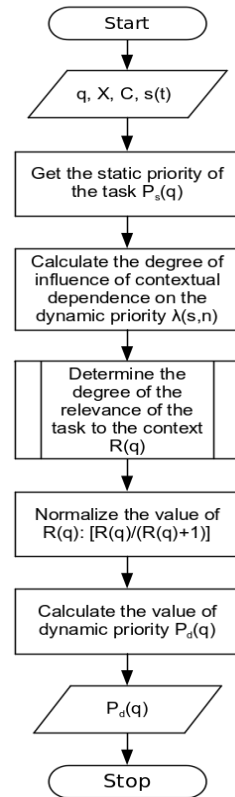


Рис. 3. Алгоритм розрахунку динамічного пріоритету завдання (A2)

4. Розрахунок динамічного пріоритету завдання

Динамічний пріоритет завдання q розраховується за формулою:

$$P_d(q) = (1 - \lambda(s,n))P_s(q) + \lambda(s,n)[R(q)/(R(q)+1)], \quad (1)$$

де $\lambda(s,n) \in [0,1]$ – ступінь впливу контекстної залежності на динамічний пріоритет, як функція від вектору контексту $s(t)$ і поточної кількості завдань n . Величиною $\lambda(s,n)$ можна регулювати силу, з якою ступінь відповідності завдання контексту впливає на місце завдання в списку Q_d . Наприклад, в розрахунку значення $\lambda(s,n)$ можна реалізувати такий принцип: чим більше завдань в Q , тим більше вплив контекстної залежності на динамічний пріоритет завдань. Крім того для специфічних векторів контексту, які еквівалентні певним екстремальним умовам роботи агента, можна встановлювати $\lambda(s,n)$ в максимальне значення. Оскільки значення ступеня відповідності завдання контексту $R(q)$ може приймати значення $R(q) > 1$, у другому доданку формули виконується його нормалізація.

Алгоритм розрахунку динамічного пріоритету (рис. 3) виконується незалежно для кожного завдання q з Q . Таким чином його програмна реалізація має великий ресурс розпаралелювання. Схема розрахунку динамічного пріоритету завдання (рис. 4) також демонструє можливість паралельної програмної реалізації алгоритму A2.

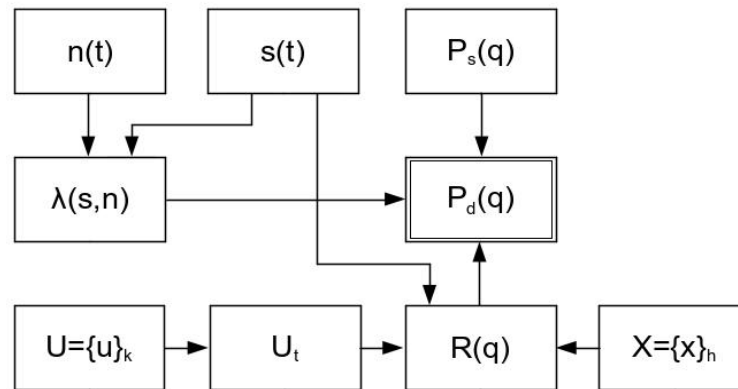


Рис. 4. Схема розрахунку динамічного пріоритету завдання

5. Визначення відповідності завдань контексту

Для визначення ступеня відповідності завдань контексту використовується множина всіх правил відповідності $U = \{u\}_k$. Кожне правило u має такий вигляд

$$u: \text{if } T_s(c) \text{ then for all } q \in Q \text{ with } T_a(x) \text{ do } R(q) = R(q) + 1, \quad (2)$$

де $T_s(c)$ – логічний вираз (Boolean expression), елементами якого є пари значень виду $c_i = v$; $T_a(x)$ – логічний вираз, елементами якого є x . Таким чином, якщо комбінація значень параметрів контексту $T_s(c)$ істинна, то для всіх завдань, для яких комбінація атрибутів $T_a(x)$ також істинна, ступінь відповідності завдання контексту буде збільшена на одиницю. Використання логічних виразів в $T_s(c)$ і $T_a(x)$ дозволяє відобразити практично будь-який аспект відповідності завдань контексту в межах обраної моделі контексту та заданого набору атрибутів завдань. Використання множини атрибутів X дозволяє задавати семантику відповідності завдань контексту. Відтак вибір атрибутів завдань та спосіб формування множини X є окремим цікавим питанням.

Ідея алгоритму визначення відповідності завдань контексту (рис. 5) полягає в тому, що чим більше правил u «спрацювало» для завдання q , тим більшою буде його ступінь відповідності контексту. При цьому для визначення відповідності завдань контексту на кроці t використовується адаптований набір правил відповідності $U_t \subseteq U$. Результатом роботи алгоритму A3 є ступінь відповідності завдання контексту $R(q)$ для кожного q .

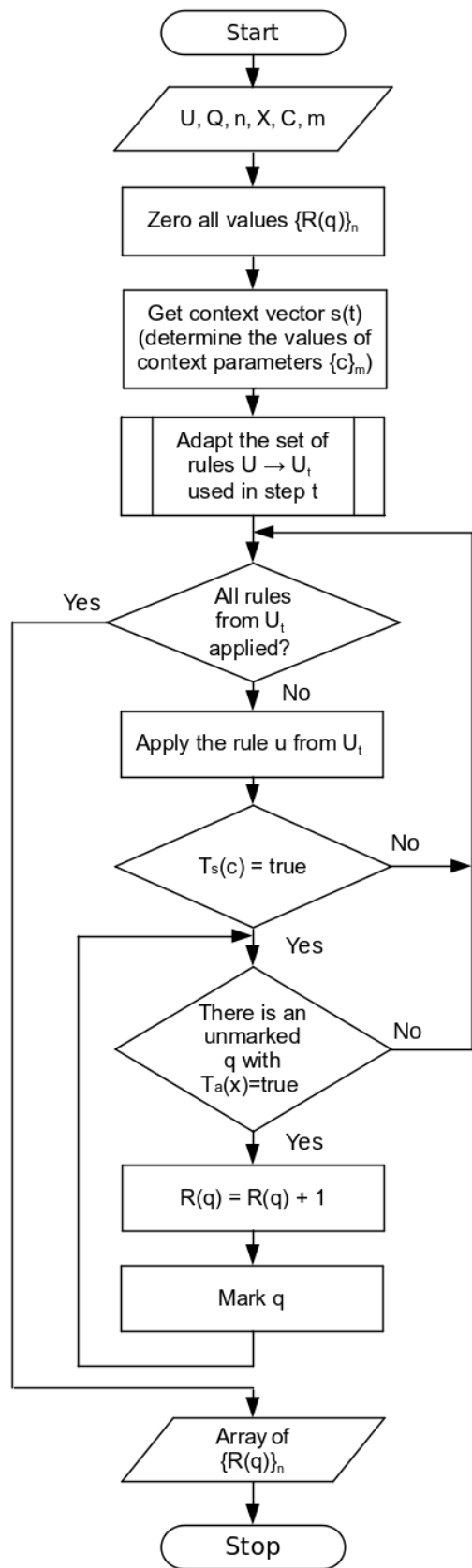


Рис. 5. Алгоритм визначення відповідності завдань контексту (A3).

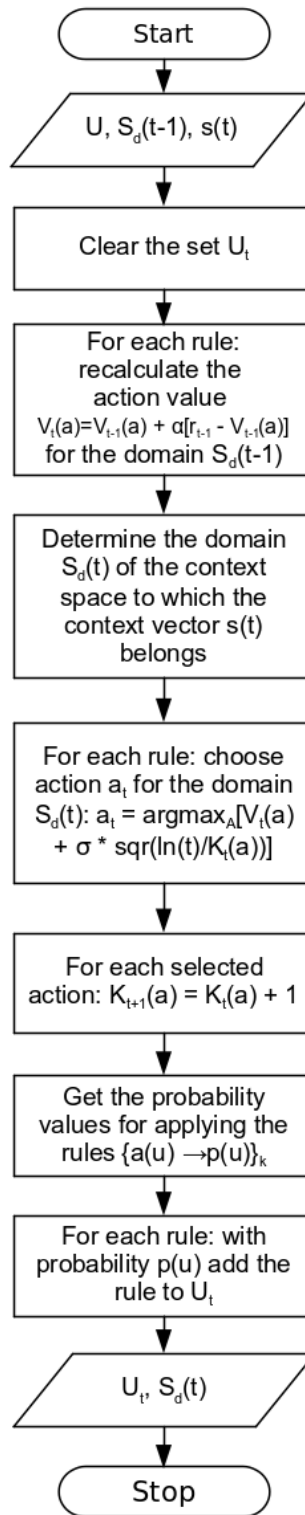


Рис. 6. Алгоритм адаптації набору правил відповідності завдань контексту (A4).

6. Адаптація набору правил відповідності завдань контексту

Проблема контекстно-залежного планування послідовності виконання завдань характеризується тим, що в більшості випадків в роботі інтелектуального агента присутні відразу кілька факторів невизначеності [1]. Це, в першу чергу, попередньо невідомі розробнику зовнішні умови роботи агента, в яких він (агент) планує послідовність виконання завдань. Додатковими факторами невизначеності є: 1) динаміка зміни множини Q (видалення одних і додавання інших завдань у «випадкові» моменти часу); 2) попередньо невідомі параметри і атрибути нових завдань, які додаються в Q ; 3) динаміка зміни множини правил відповідності U з боку зовнішніх по відношенню до агента сутностей та ряд інших чинників. Вплив факторів невизначеності можна зменшити шляхом адаптації набору правил відповідності U до зовнішніх умов роботи інтелектуального агента і частково до «внутрішніх» змін, спираючись на оцінку ефективності роботи агента. Одним з найбільш перспективних підходів в даному випадку є використання машинного навчання, зокрема методів навчання з підкріпленням [14,15].

У даній статті для адаптації набору правил відповідності завдань контексту використана схема навчання з підкріпленням в стаціонарному випадковому середовищі з контекстною залежністю (contextual multi-armed bandit problem) [14]. Ця схема застосовується окремо для кожного правила u , тобто для кожного правила u виконується свій примірник процедури навчання з підкріпленням (всього k примірників). В якості контексту для всіх примірників процедури навчання з підкріпленням розглядається загальний контекст S та відповідний простір контексту (S, d) . За допомогою метрики d простір контексту розбивається на домени $\{S_d\}_M$. Два вектори контексту s_i та s_j потрапляють в один домен за умови, що

$$d(s_i, s_j) = \|s_i - s_j\| < \mu, \mu > 0. \quad (3)$$

Величина μ визначає розмір окремого домену і відповідно загальну кількість доменів M . Процедура навчання з підкріпленням для кожного правила u додатково розбивається на M примірників за кількістю доменів. Таким чином для кожного домену S_d та правила u виконується свій примірник процедури навчання з підкріпленням (всього kM примірників).

Схема навчання з підкріпленням полягає в наступному (рис. 6). Для кожного правила відповідності u визначено множини дій $A = \{a_i\}_L$. Кожна дія a_i визначає ймовірність застосування правила u на кроці t (тобто правило u з множини U відбирається в множини U_t на кроці t_c з ймовірністю $p_i(u)$):

$$a_i: p_i(u) = (i-1) / (L-1), i = 1, \dots, L. \quad (4)$$

В якості підкріплення r_t за вибір дії a_i на кроці t використовується оцінка ефективності роботи агента, отримана для поточного списку актуальних завдань Q_A , сформованого на підставі адаптованого набору правил U_t . Вибір дії a_i визначає частоту застосування правила u у відповідному домені простору контексту (S, d) . Використання процедури навчання з підкріпленням дозволяє знаходити таку частоту застосування правила u в домені S_d , яка максимізує середнє значення оцінки ефективності роботи агента для пари (u, S_d) .

У кожному примірнику процедури навчання з підкріпленням для кожної обраної дії перераховується її оціночна вага (action value):

$$V_{t+1}(a) = V_t(a) + \alpha[r_t - V_t(a)], \quad (5)$$

де $\alpha = \text{const}$ – крок навчання, такий що $\alpha \in (0, 1]$. Для вибору наступної дії використовується метод верхньої довірчої межі (Upper-Confidence-Bound Action Selection, UCB) [14]:

$$a_t = \text{argmax}_A [V_t(a) + \sigma \cdot \text{sqr}(\ln(t) / K_t(a))], \quad (6)$$

де $\sigma > 0$ – параметр, який визначає ступінь дослідження (degree of exploration), $K_t(a)$ – кількість разів, коли було обрано дію a , на момент часу t .

Схема роботи процедури навчання з підкріпленням (рис. 7), реалізована в прототипі модуля планування послідовності виконання завдань, дозволяє використовувати декомпозицію даних $(S_d, \{V(a)\})$ і функціональну декомпозицію ($[\text{select } S_d], [\text{argmax}], [\text{calculate } V(a)]$) для розпаралелювання відповідних обчислень.

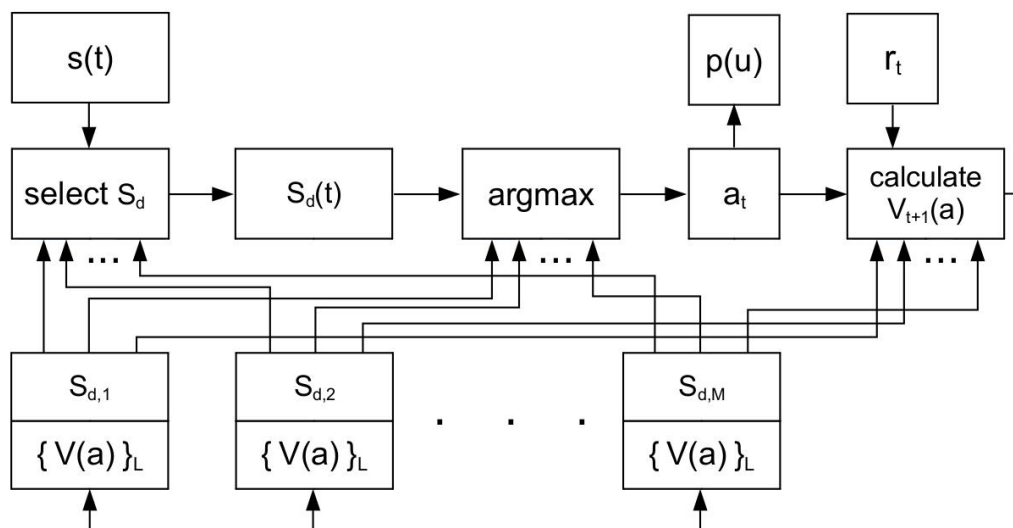


Рис. 7. Схема роботи процедури навчання з підкріпленням

Висновки

Розглянуто проблему контекстно-залежного планування послідовності виконання незалежних або слабпов'язаних завдань інтелектуальним агентом. Проаналізовано принцип відповідності завдання контексту. Запропоновано структуру модуля контекстно-залежного планування послідовності виконання завдань у складі інтелектуального агента та алгоритм його роботи.

Також в статті запропоновано алгоритм розрахунку динамічного пріоритету завдання, алгоритм визначення відповідності завдання контексту і алгоритм адаптації набору правил відповідності завдань контексту на основі навчання з підкріпленням в стаціонарному випадковому середовищі з контекстною залежністю (contextual multi-armed bandit problem). Запропоновано схему роботи процедури навчання з підкріпленням, реалізовану в прототипі модуля контекстно-залежного планування послідовності виконання завдань інтелектуальним агентом. Отримані результати можуть бути узагальнені на випадок планування послідовності виконання завдань багатоагентною системою, в тому числі з урахуванням розподілу окремих завдань між агентами.

Список літератури

1. Stuart Russell, Peter Norvig, *Artificial Intelligence: A Modern Approach*, 4th edition, Pearson, 2020. – 1136 p. ISBN: 978-0-13-461099-3
2. David L. Poole, Alan K. Mackworth, *Artificial Intelligence: Foundations of Computational Agents*, 2nd Ed., Cambridge University Press, 2017. – 820 p. ISBN: 978-1-13-964321-4
3. *Multiagent Systems*, by Gerhard Weiss (Editor), 2nd Ed., The MIT Press, 2013. – 920 p. ISBN: 978-0-26-253387-4
4. Tierney K., Browne J., *Task sequence planning*, In: Bernhardt R., Dillman R., Hörmann K., Tierney K. (eds) *Integration of Robots into CIM*. Springer, 1992. – pp. 36-44, ISBN: 978-0-41-237140-0
5. Tiehua Cao, Arthur C. Sanderson, *Intelligent Task Planning Using Fuzzy Petri Nets*, World Scientific, 1996 – 192 p. ISBN: 978-9-81-022556-8
6. J. Rosell, N. Munoz and A. Gambin, "Robot tasks sequence planning using Petri nets," *Proceedings of the IEEE International Symposium on Assembly and Task Planning*, 2003., Besancon, France, 2003, pp. 24-29, DOI: 10.1109/isatp.2003.1217182
7. L. Zhang et al., "Adaptive quantum genetic algorithm for task sequence planning of complex assembly systems," in *Electronics Letters*, vol. 54, no. 14, 2018. – pp. 870-872, DOI: 10.1049/el.2018.0609
8. Schilit B., Adams N., Want R., *Context-aware computing applications*, in *Proceedings of the IEEE Workshop on "Mobile Computing Systems and Applications"*, IEEE Computer Society, 1994. – pp. 85-90, DOI: 10.1109/wmcsa.1994.16

9. Abowd G.D., Dey A.K., Brown P.J., Davies N., Smith M., Steggle P. (1999) *Towards a Better Understanding of Context and Context-Awareness*. In: Gellersen H.W. (eds) *Handheld and Ubiquitous Computing. HUC 1999. Lecture Notes in Computer Science*, vol 1707. Springer, Berlin, Heidelberg. – pp. 304-307, DOI: 10.1007/3-540-48157-5_29

10. Cristiana Bolchini, Carlo A. Curino, Elisa Quintarelli, Fabio A. Schreiber, and Letizia Tanca. 2007. *A data-oriented survey of context models*. *ACM SIGMOD Record*, 36, 4 (December 2007), pp. 19-26, DOI: 10.1145/1361348.1361353

11. C. Perera, A. Zaslavsky, P. Christen and D. Georgakopoulos, "Context Aware Computing for The Internet of Things: A Survey," in *IEEE Communications Surveys & Tutorials*, vol. 16, no. 1, First Quarter 2014, pp. 414-454, DOI: 10.1109/surv.2013.042313.00197

12. Grifoni P., D'Ulizia A., Ferri F., *Context-Awareness in Location Based Services in the Big Data Era*, In: Skourletopoulos G., Mastorakis G., Mavromoustakis C., Dobre C., Pallis E. (eds) *Mobile Big Data. Lecture Notes on Data Engineering and Communications Technologies*, Springer, vol 10, 2018. – pp. 85–127, DOI: 10.1007/978-3-319-67925-9_5

13. Nicholas Capurso, Bo Mei, Tianyi Song, Xiuzhen Cheng, *A survey on key fields of context awareness for mobile devices*. *Journal of Network and Computer Applications*, Volume 118, 2018. – pp. 44-60, DOI: 10.1016/j.jnca.2018.05.006

14. Richard S. Sutton, Andrew G. Barto, *Reinforcement Learning: An Introduction*, 2nd Ed., A Bradford Book, 2018. – 532 p. ISBN: 978-0-26-203924-6

15. Csaba Szepesvari, *Algorithms for Reinforcement Learning*, Morgan and Claypool Publishers, 2010. – 104 p. ISBN: 978-1-60-845492-1

TASK SEQUENCE PLANNING BY INTELLIGENT AGENT WITH CONTEXT AWARENESS

A. Botchkaryov

Lviv Polytechnic National University,
Computer Engineering Department

© Botchkaryov A., 2022

The problem of context-aware task sequence planning for independent or weakly related tasks by an intelligent agent has been considered. The principle of matching the task to the context is analyzed. The structure of the context-aware task sequence planning module as part of an intelligent agent and the corresponding algorithm are proposed. In the structure of the planning module, three main groups of blocks are implemented: operations with tasks, operations with the context, determining the relevance of tasks to the context.

The article also proposes an algorithm for calculating the dynamic priority of a task, an algorithm for determining the relevance of a task to the context, and an algorithm for adapting a set of rules for matching the task to the context. The value of the dynamic priority depends on the static priority of the task, which is assigned when a new task is added, and the degree of correspondence of the task to the context, taking into account the context vector. Two modes of starting the planning algorithm are allowed: when the intelligent agent requests a new task and when the context vector changes. The dynamic priority calculation algorithm is performed independently for each task. As a result, its software implementation has a large parallelization resource.

To adapt the set of rules for matching tasks to the context, a scheme of reinforcement learning in a contextual multi-armed bandit was used. For each matching rule, a separate instance of the reinforcement learning procedure is performed. The reinforcement learning method used in the article is Upper-Confidence-Bound Action Selection. The functional scheme of the reinforcement learning procedure, implemented in the prototype of the context-aware task sequence planning module has been proposed. The functional scheme of the reinforcement learning procedure allows the use of data decomposition and functional decomposition to parallelize the corresponding calculations.

Key words: context awareness, intelligent agent, task sequence planning.