



РОЗРОБКА ТА ДОСЛІДЖЕННЯ МУЛЬТИАГЕНТНОЇ СИСТЕМИ ДЛЯ ОХОРОНИ ОБ'ЄКТА

П.М. Тишин¹, А.В. Гончаров², К.О. Куширець³

^{1,2,3}Одеський національний політехнічний університет, Україна

E-mail: ¹petrmettal@gmail.com, ²orlandmax@gmail.com, ³almag336@gmail.com.

Copyright © 2017 by author and the journal "Automation technological and business - processes".

This work is licensed under the Creative Commons Attribution International License (CC BY).

<http://creativecommons.org/licenses/by/4.0/>



DOI: 10.15673/atbp.v10i4.825

Анотація: У статті описується мультиагентна система на основі онтології OWL, враховуючи стандарти FIPA для системи охорони об'єкта. Наведено дані про роботу інтелектуальних агентів і комунікації між ними, а також пропозиції для вирішення поставлених завдань. Запропонована концепція дозволяє значно скоротити кількість електроенергії, що витрачається в процесі роботи системи стеження за об'єктами на певному периметрі з метою охорони об'єкта, знизити ціну на вартість системи, підвищити її ефективність, а також забезпечити легку масштабованість. Крім реалізації самої мультиагентної системи, були вирішені додаткові завдання - реалізовані алгоритми побудови передбачуваного маршруту руху об'єкта, вибору камери для спостереження за об'єктом, діагностування системи на наявність помилок. Завдяки особливостям взаємодії інтелектуальних агентів ця система підлягає простому розширенню в разі збільшення території спостереження або збільшенні кількості датчиків на певній території. Структурно охоронна система складається з центру управління з системою контролю, де сидить оператор, що стежить за периметром навколо об'єкту, що охороняється; набору мікрокомп'ютерів, що стежать за певним сектором території, що охороняється за допомогою датчиків (основних і допоміжних); поворотних камер, що забезпечують візуальний огляд околиць об'єкта. Ефективний алгоритм комунікації і набір правил для агентів дозволяє захопити максимальну кількість об'єктів, використовуючи колективні ресурси агентів камер. Для запобігання виходу з ладу елементів охоронної системи передбачена система контролю, яка в свою чергу управляється інтелектуальним агентом, який взаємодіє з агентами датчиків і агентом підтримки прийняття рішень. Надалі результати дослідження можуть бути використані для удосконалення систем охорони об'єктів.

Abstract: The article describes a multi-agent system based on the OWL ontology, taking into account the FIPA standards for the object security system. The given data on the work of intellectual agents and communication between them, as well as proposals for solving the tasks assigned. The proposed concept allows to significantly reduce the amount of electricity consumed during the operation of the object tracking system at a certain perimeter for the purpose of object protection. The range of tasks to be solved is not limited only to the implementation of the multi-agent system, among other things, such algorithms as constructing the proposed route of the object's movement, selecting a camera for tracking the object, diagnosing the system for errors are realized. Due to the peculiarities of the interaction of intelligent agents, this system is subject to simple expansion in the event of an increase in the surveillance area or an increase in the number of sensors in a certain territory. Structurally, the security system consists of a control center with a control system, where an operator sits observing the perimeter around the protected object; a set of microcomputers monitoring a certain sector of the protected area using sensors (main and auxiliary); two rotary cameras providing a visual inspection of the surroundings of the facility. An effective communication algorithm and a set of rules for agents allows you to capture the maximum number of objects using the shared resources of agents of two cameras. To prevent the failure of the elements of the security system, a monitoring system is provided, which in turn is controlled by an intelligent agent that interacts with the sensor agents and the decision support agent. In the future, the results of the study can be used to improve the protection of objects.

Ключові слова: мультиагентна система, онтологія OWL, охорона об'єкту, енергоефективність, агентно-орієнтований підхід.

Keywords: multi-agent systems, ontology OWL, object protection, energy efficiency, agent-based approach.

Вступ

Протягом останніх чотирьох років в Україні зростає рівень злочинності, ця проблема стосується і крадіжок з приватних будинків. Рішенням цієї проблеми може стати використання інтелектуальних охоронних систем. Це дозволить розвантажити оператора-охоронця шляхом часткового перенесення функцій з на мультиагентну систему.



Важливо, щоб при реалізації такої системи, особлива увага приділялася критерієм енергоефективності, що враховано при її проектуванні.

Звернемося до огляду існуючих систем охорони об'єктів, щоб визначити їх недоліки та на базі цих знань запропонувати конкурентоспроможну концепцію системи для реалізації поставленого завдання. Методики рішення досліджуваної проблеми можна поділити на дві категорії - з використанням мультиагентних систем і без. Системи охорони об'єктів, які не використовують мультиагентні системи використовують такі програмно-апаратні моделі:

1. Датчики по периметру і камери (одна або кілька).

Цей підхід найбільш поширений, оскільки досить дешевий і простий в реалізації. Істотна економія досягається шляхом скорочення кількості датчиків. Однак це знижує кількість відомих даних про об'єкт, складніше прогнозувати його дії, всі контрольні функції покладені на камери, які працюють окремо або незначно взаємодіють між собою через комп'ютер, завдяки клієнт-серверній архітектурі програми, що управляє системою. Існує значна кількість робіт, які спрямовані на те, щоб підвищити ефективність роботи подібної системи без збільшення витрат на обладнання.

2. Кілька камер без датчиків.

Ця реалізація близька до попереднього підходу, але робить неможливим економію енергії. Є одним з найбільш економічних, однак малоефективних варіантів.

3. Кілька датчиків без камер.

Найбільш економічний варіант, який значним чином звужує область його застосування. Його можна використовувати в не сильно людних місцях, коли об'єкт, що охороняється відвідують досить рідко, це дозволяє мінімізувати хибнопозитивні спрацьовування такого підходу. Потрібна наявність оператора для перевірки периметра в разі спрацьовування датчика.

Якщо розглядати мультиагентні системи для вирішення цього завдання, то найбільший інтерес представляють концепції, запропоновані в статтях «Coopetitive multi-camera surveillance using model predictive control», авторства Вівек Сингх, Моханом Канканхаллі, Прадіпом Атрі [8], а також в статті «Agent-based coordination of cameras» Гарсії І., Карбо Х., Моліні М. [6]. Розглянемо запропоновані ними рішення більш детально. У статті «Competitive multi-camera surveillance using model predictive control» обґрунтовується необхідність обміну всіма даними між компонентами системи для побудови ефективної превентивної стратегії роботи. Перевага цього підходу полягає в можливості зменшення затримок в реакції системи, отриманні більш чіткого зображення завдяки використанні декількох камер для спостереження за одним об'єктом. Проте, у зв'язку з необхідністю безперервно обмінюватися даними між всіма компонентами системи, всі вони повинні бути включені, що робить неможливим перехід елементів системи до режиму економії електроенергії, а отже вимагає значних витрат, крім того, такий підхід вимагає значно більших обчислювальних можливостей, ніж інші традиційні методи вирішення проблеми охорони об'єкта. Стаття «Agent-based coordination of cameras» пропонує наступний підхід з використанням мультиагентної системи: відцентрувати напрямок огляду камери для уточнення положення об'єкта шляхом, передачі даних між агентами камер і датчиків. Недоліками такого підходу є необхідність збільшення кількості датчиків для збільшення точності результатів, здорожує реалізацію системи. Алгоритм роботи цієї системи не передбачає можливість самовідновлення або розширення, також накладає певні обмеження для її успішної безперервної роботи.

Мета роботи

Розробка та дослідження інтелектуальних агентів і їх комунікації в мультиагентній системі на базі стандарту FIPA для охорони об'єктів.

Основна частина

У зв'язку з виявленими недоліками можна запропонувати наступне рішення. Програмно-апаратна платформа для відеоспостереження з камерами і датчиками по периметру, яка буде управлятися мультиагентною системою, з наступними характеристиками:

1. максимальне зменшення витрат на електроенергію для підтримки життєдіяльності побудованої системи;
2. відстеження значно більшого числа об'єктів, що одночасно знаходяться в полі зору системи камер;
3. прогнозування напрямку руху об'єктів, завдяки інтелектуальності датчиків і їх взаємодії між собою;
4. аналіз подій, яке відбувається в поле зору системи і підтримка прийняття рішень;
5. автоматизація розширення системи датчиків і реалізація самовідновлення функцій системи в разі виходу з ладу певних її елементів без втручання оператора.

Порівняння рішень наведені у Таблиці 1.

Перейдемо безпосередньо до опису розробленого апаратно-програмного комплексу. Для реалізації повноцінної системи для охорони об'єкта потрібно реалізувати наступні інтелектуальні агенти:

1. Агенти-датчики.
2. Агенти-камери.
3. Агент системи контролю.
4. Агент підтримки прийняття рішень.

Датчики, якими керують агенти-датчики, на апаратному рівні складаються з мікрохвильового датчика руху, теплового датчика руху, датчика енергоспоживання та мікроконтролера з підтримкою CAN.



Таблиця 1 – Порівняння рішень для вирішення завдання охорони об'єкта

Порівняння рішень охорони об'єкта	Ціна	Енергоефективність	Ефективність	Розширюваність
Не мультиагентний підхід				
Тільки датчики	дуже низька	дуже висока	дуже низька	не потрібна
Тільки камери	низька	низька	низька	середня
Камери та датчики	середня	низька	середня	середня
Мультиагентний підхід				
Камери та датчики	середня	низька	висока	низька
Надлишковий обмін даними	висока	дуже низька	дуже висока	низька
Предлагаемый подход	середня	висока	висока	висока

Мікрохвильовий датчик руху налаштований на довжину хвилі, яка збігається з температурою тіла людини. Це дозволяє не фіксувати проходження різних тварин, в свою чергу зменшує навантаження на систему і оператора. Крім того, датчики такого типу відрізняються підвищеною надійністю, завдяки можливості працювати через прозорий для радіохвиль полімер, який може бути використаний для захисту від опадів і вандалізму. Наступний елемент системи - тепловий датчик руху, який налаштований на фіксування тепла, яке відповідає рухомий машині в тому випадку, коли зловмисник використовує транспорт і мікрохвильовий датчик руху не може зафіксувати його присутність поблизу або всередині периметра об'єкту, що охороняється. Третій датчик вимірює енергоспоживання датчиків для контролю їх роботи. Якщо в деякий проміжок часу енергоспоживання датчиків зростає і перевищила допустимий рівень, це означає ймовірність виходу з ладу системи. У такому випадку потрібно відключити датчики від мережі енергоспоживання і повідомити це система контролю за системою. Останній елемент системи - мікроконтролер з апаратною підтримкою CAN-протоколу який приймає повідомлення від програмної оболонки інтелектуального агента, а також виконує збір, первинну обробку та передачу інформації по датчиків руху на енергоживлення. Вибір CAN визначається наступним набором можливостей: пріоритетне вирішення конфліктної ситуації при зверненні до шини, виявлення помилок і їх усунення на апаратному рівні, автоматичну ретрансляцію пакетів, раніше переданих з помилкою, автоматичне відключення несправних вузлів. Таким чином на апаратному рівні забезпечується надійна передача інформації від кожного датчика до агенту системи.

Датчики об'єднуються в єдиний ланцюг, який називається сектором. Кожен сектор розбивається на три підсектори:

1. підсекторів граничних датчиків. Датчики в цьому секторі сигналізують про те, що людина знаходиться поблизу периметра об'єкту, що охороняється. Стандартний режим - активний. Можуть бути замінені звичайним датчиками в разі виходу з ладу. Місцезнаходження підсектори - периметр об'єкта, що охороняється.
2. підсекторів тривожних датчиків - датчики сигналізують про те, що людина знаходиться в безпосередній близькості від об'єкту, що охороняється. Стандартний режим - сон. Також можуть бути замінені звичайними датчиками в разі виходу з ладу. Місцезнаходження підсектори - безпосередньо поблизу до об'єктів.
3. підсекторів звичайних датчиків. Стандартний режим - сон. Служить допоміжною ланкою в разі виявлення об'єкта по периметру. Грає роль резервного ресурсу в разі виходу з ладу тривожних або граничних датчиків. Місцезнаходження підсектори - територія між тривожними і граничними датчиками.

Залежно від того, який розмір охоплює територія навколо об'єкту, що охороняється, довжина секторів і кількість датчиків можуть змінюватися.

Пристрої передачі зображення, в залежності від умов, на апаратному рівні може бути представлені тепловізорами, пристроями нічного бачення, звичайними камерами і штативом, який обертається на максимальний кут 360 градусів. Управління штативом здійснюється через CAN шини. Передача зображення на пост оператора забезпечується за допомогою Ethernet кабелю. Площа навколо об'єкту, що охороняється, представляється у вигляді кола, яке розбивається на 8 секторів. Охоронна система складається з двох камер, кожна з яких по-замовчанню відповідає за 4 сектори. У разі необхідності камери можуть переходити на сектора іншої камери, в разі необхідності (кілька об'єктів потрапили в сектора однієї камери). Пристрої передачі зображення можуть обертатися в двох площинах. Це забезпечує практично повне покриття спостерігається площі.



Агент системи контролю створений з метою моніторингу системи на предмет виявлення помилок в її роботі і приймає рішення про подальше їх виправлення. На самому початку роботи агент формує карту датчиків, де зберігається інформація про кожний датчик: тип датчика (звичайний, граничний, тривожний), його стан (сплячий режим, активний режим, датчик зламанний). Через однаковий проміжок часу система розсилає запит про надання з кожного датчика актуальної інформації про його стан і оновлює карту датчиків. Наприклад, в проміжок між двома запитами датчик, вийшов з ладу, був замінений на новий. Система автоматично відзначає це на карті датчиків. Агент підтримки прийняття рішень збирає дані з камер і агента системи контролю та аналізу, на основі яких працює система підтримки прийняття рішень.

Програмна реалізація представлена в як фреймворка JADE для програми IntelliJ IDEA, де вже існує зручна система обміну транспортними повідомленнями між агентами, реалізований зручний інтерфейс. Для аналізу роботи програм, реалізованих за допомогою фреймворка JADE, можна використовувати діаграми класів, діаграми варіантів використання (UML), діаграми послідовності. Завдяки спеціальним можливостям, влаштованим в JADE більшість таких діаграм може бути побудовано в автоматичному режимі. Це дозволяє удосконалювати алгоритм, через наочність отриманих даних. Крім того, вищевказаний функціонал повністю відповідає стандарту FIPA [9], що необхідно для побудови якісної архітектури програми. Для реалізації цілісного алгоритму роботи мультиагентної системи побудована онтологія OWL (база знань), якою оперують агенти в системі. База знань може бути реалізована як у формі бази даних, так і в формі RDFS документа, схожого за своїм синтаксисом на XML, але створеного спеціально для обміну даними між інтелектуальними агентами. В ході проведення аналізу було вирішено реалізовувати базу знань OWL за допомогою мови RDFS [7]. Від реалізації за допомогою баз даних вирішено було відмовитися, оскільки виконання рекурсивних запитів вимагає великих обчислювальних витрат. Тяжкість їх реалізації призводить до необхідності вибрати підхід, який повністю відповідає стандарту FIPA для мультиагентних систем, але не вимагає побудови вищезгаданих властивостей.

Витяг з онтології OWL описаних на мові RDFS для охорони об'єкта.

```
<owl:Ontology rdf:about="">
<rdfs:comment">Онтологія системи охорони території</rdfs:comment>
</owl:Ontology>
//Основні класи
<owl:Class rdf:ID="Камера" >
<owl:Restriction>
<owl:onProperty rdf:resource="#Має" />
<owl:someValuesFrom rdf:resource="#КутПовороту" />
</owl:Restriction>
<owl:Restriction>
<owl:onProperty rdf:resource="#Має" />
<owl:someValuesFrom rdf:resource="#ШвидкістьПовороту" />
</owl:Restriction>
</owl:Class>
<owl:Class rdf:ID="ДатчикПуху" />
<owl:Restriction>
<owl:onProperty rdf:resource="#ЗнаходитьсяСтані" />
<owl:someValuesFrom rdf:resource="#СтанАктивності" />
</owl:Restriction>
</owl:Class>
<owl:Class rdf:ID="КоординатаX" />
<owl:Class rdf:ID="КоординатаY" />
<owl:Class rdf:ID="СтанСистеми" />
<owl:Class rdf:ID="СпостерігаємийОб'єкт" />
<owl:Class rdf:ID="КутПовороту"> // поточний кут повороту камер 360.
<rdfs:minCardinality rdf:datatype="&xsd;nonNegativeInteger">0</rdfs:minCardinality>
<rdfs:maxCardinality rdf:datatype="&xsd;nonNegativeInteger">360</rdfs:maxCardinality>
</owl:Class>
<owl:Class rdf:ID="ШвидкістьПовороту" > // поточна швидкість повороту камер
<rdfs:minCardinality rdf:datatype="&xsd;float">0.00</rdfs:minCardinality>
<rdfs:maxCardinality rdf:datatype="&xsd;float">1.00</rdfs:maxCardinality>
</owl:Class>
<owl:Class rdf:ID="СтанАктивності">
// Підкласи
<owl:Class rdf:ID="ГраничнийДатчик">
<rdfs:subClassOf rdf:resource="#ДатчикДвижения" />
</owl:Class>
```



```
<owl:Class rdf:ID="НеграничнийДатчик">
<rdfs:subClassOf rdf:resource="#ДатчикДвижения"/>
<owl:complementOf rdf:resource="#ГраничныйДатчик"/> // ограничение по принадлежности
</owl:Class>
<owl:Class rdf:ID="ТривожнийДатчик">
<rdfs:subClassOf rdf:resource="#НеграничнийДатчик">
</owl:Class>
// Індивіди класів для класу СтанСистеми
<СостояниеСистемы rdf:ID="Очікування"/>
<СостояниеСистемы rdf:ID="Спостереження"/>
<СостояниеСистемы rdf:ID="Тривога"/>
// Індивіди класів для класу ДатчикРуху
<ДатчикРух rdf:ID="ДатчикРух№1"/>
<ДатчикРух rdf:ID="ДатчикРух№2"/>
<ДатчикРух rdf:ID="ДатчикРух№3"/>
...
<ДатчикРух rdf:ID="ДатчикРух№520"/>
// Індивіди класів для класу Камера
<Камера rdf:ID="CAM1"/>
<Камера rdf:ID="CAM2"/>
// Індивіди класів для класу СтанАктивності
<СтанАктивності rdf:ID="Робота"/>
<СтанАктивності rdf:ID="Сон"/>
<СтанАктивності rdf:ID="Несправність"/>
// Опис властивостей для класу ДатчикРуху
<owl:ObjectProperty rdf:ID="МаєКоординатуХ">
<rdfs:domain rdf:resource="#ДатчикРуху"/>
<rdfs:range rdf:resource="#КоординатаХ"/>
</owl:ObjectProperty>
<owl:ObjectProperty rdf:ID="МаєКоординатуY">
<rdfs:domain rdf:resource="#ДатчикРуху"/>
<rdfs:range rdf:resource="#КоординатаY"/>
</owl:ObjectProperty>
<owl:ObjectProperty rdf:ID="ЗнаходитьсявСтані">
<rdfs:domain rdf:resource="#ДатчикРуху"/>
</owl:ObjectProperty>
// Опису властивостей класу Камера
<owl:ObjectProperty rdf:ID="Має">
<rdfs:domain rdf:resource="#Камера"/>
<rdfs:range rdf:resource="#КутПовороту"/>
</owl:ObjectProperty>
<owl:ObjectProperty rdf:ID="Має">
<rdfs:domain rdf:resource="#Камера"/>
<rdfs:range rdf:resource="#ШвидкістьПовороту"/>
</owl:ObjectProperty>
// Опис координат для індивідів класу
<ДатчикРуху№1>
<МаєКоординатуХ rdf:resource="#2,5"/>
<МаєКоординатуY rfd:resource="#2,5"/>
</ДатчикРуху №1>
...
<ДатчикРуху №520>
<МаєКоординатуХ rdf:resource="#5"/>
<МаєКоординатуY rfd:resource="#2,5"/>
</ДатчикРуху №520>
```

Використані інтелектуальні агенти являють собою BDI-агентів. Наведемо їх таким чином, щоб вони відповідали підходу, запропонованому у роботі Д.Ю. Бугайченко [1].

Специфікація можливих дій BDI-агентів:

1. Агент управління датчиками SD з можливими діями:

- включити датчик (sensor_on) - об'єкт агента датчика включає датчик з відповідним id на зазначеному секторі



- вимкнути датчик (sensor_off) - об'єкт агента датчика вимикає датчик з відповідним id на зазначеному секторі
 - передати показання (send_sensor_data) - об'єкт агента датчика передає дані системі, які включають в себе енергоживлення датчика, дані з мікрохвильового датчика руху, дані з теплового датчика руху.
 - нічого не робити (noopSD) - в разі отримання повідомлення об'єкт агента датчика не виконує ніяких дій.
2. Агент управління камерами CD з можливими діями:
- включити камеру (camera_on) - об'єкт агента-камери включає камеру з відповідним id
 - камеру (camera_off) - об'єкт агента-камери вимикає камеру з відповідним id.
 - передати показання (send_camera_data) - об'єкт агента-камери з відповідним id передає інформацію про конфліктну ситуацію, поточний кут повороту до мети, пріоритет спостережуваного об'єкта.
 - прийняти показання координат об'єктів (receive_coordinates) - об'єкт агента-камери з відповідним id приймає координати ділянок за якими потрібно вести спостереження, пріоритети цілей, поточний кут повороту і пріоритет спостережуваного об'єкта іншу камеру.
 - передати показання координат об'єктів (send_coordinates) - об'єкт агента-камери з відповідним id передає координати спостережуваних ділянок
 - нічого не робити (noopCD) - в разі отримання повідомлення об'єкт агента-камери не виконує ніяких дій.
3. Агент системи контролю CS з можливими діями:
- сформувати карту датчиків (init_sensors_map) - об'єкт агента системи контролю ініціює нову карту датчиків.
 - редагувати карту датчиків (edit_sensors_map) - об'єкт агента системи контролю редагує передане стан датчика на карті.
 - передати карту датчиків (send_sensors_map) - об'єкт агента системи контролю передає карту датчиків.
 - нічого не робити (noopCS) - в разі отримання повідомлення об'єкт агента системи контролю не виконує ніяких дій.
4. Агент системи підтримки прийняття рішень DS з можливими діями:
- сформувати пріоритет мети (form_priority) - об'єкт агента системи прийняття рішень формує пріоритет за поточною метою.
 - дати оцінку ситуації (get_analyze_current_situation) - об'єкт агента системи прийняття рішень передає оператору інструкції на основі поточного стану навколишнього середовища.
 - нічого не робити (noopCS) - в разі отримання повідомлення об'єкт агента системи прийняття рішень не виконує ніяких дій.

Під картою датчиків в третьому агента мається на увазі двовимірний масив, який зберігає координати всіх датчиків, які перебувають на території, що охороняється, їх тип (звичайний, граничний, тривожний), а також стан (включений, вимкнений, пошкоджений).

Всі ці команди реалізовані на програмному рівні і повинні служити зручним способом зв'язку користувача (оператора-охоронця) з програмним забезпеченням розробленої системи (мультиагентної системи охорони об'єкта).

Для розробки алгоритмів взаємодії інтелектуальних агентів в системі застосовувався підхід, аналогічний рішенням [2,3,4,5], у яких вводяться поняття мережі потреб та можливостей, де кожен агент має певний ресурс і правильне розподілу цих ресурсів в мультиагентної системі має найголовніше значення. Серед методів роботи з даною мережею для вирішення завдання передбачення можливих маршрутів руху об'єктів, а також алгоритму вибору камери для спостереження за об'єктом на базі набору критеріїв особливий інтерес представляють методи віртуального «круглого столу» і «змінного» прийняття рішень.

У мережі реалізовані наступні алгоритми: вибір камери для спостереження за об'єктом, побудова ймовірного маршруту руху об'єкта, вирішення проблеми виходу з ладу датчика, комунікація між оператором та агентами через агента підтримки прийняття рішень.

1. Процес вибору камери для спостереження за об'єктом. Для цього використовується набір критеріїв: різниця в відстані в градусах до об'єкту між камерами, зайнятість кожної з камер, пріоритет об'єкта, який вимагає стеження, пріоритети об'єктів, які спостерігаються обома камерами, сектор положення об'єкта (частина секторів за замовчуванням належать до камери 1, інша – до камери 2). Такі параметри як пріоритет і зайнятість камер є якісними, тобто їх можна уявити за допомогою нечіткої логіки і проводити операції над цими величинами в межах цього математичного апарату, інші характеристики приймають чіткі значення, тому для обчислень з ними можна використовувати звичайну логіку. Цей набір критеріїв є значення змінних x_0 - x_7 . Завдяки логічних операцій «I» і ставленням «БІЛЬШЕ», «МЕНШЕ», «РІВНО» при аналізі поточного стану критеріїв можна зробити в один з висновків про вибір камери для спостереження за новим об'єктом. Пріоритет камера отримує завдяки аналізу об'єкта агентом підтримки прийняття рішень. Можливо п'ять варіантів розвитку подій - камера 1 повинна почати спостереження, камера 2 повинна почати спостереження, жодна з камер не повинна стежити за новим об'єктом, обидві камери повинні почати відстеження об'єкта, вибір камери повинен зробити оператор. В математичному апараті цей набір варіантів розвитку події являє собою набір вихідних змінних y_0 - y_5 . Завдяки розробленому набору правил для дій після обміну даними між камерами після аналізу вхідних даних (X) отримуємо одне з вихідних даних (Y). Що, зрештою, і потрібно для реалізації цього завдання.

2. Побудова ймовірного маршруту руху об'єкта. Через кожен момент часу Δt формується карта перебування



об'єктів, шляхом отримання інформації від датчиків. Карти, які представляють собою двовимірні масиви порівнюються між собою і звідси робиться висновок про можливі траєкторії руху. Ймовірні стану представлені у вигляді списку точок, де був і куди прямує об'єкт. Якщо спостережуваний об'єкт змінює напрямку руху, то траєкторії, що не відповідають поточним даними, викреслюються з бази знань агента. Якщо приблизний маршрут руху стосується одного з об'єктів, які необхідно захищати, то агент камери повідомляє про це агентів підтримки прийняття рішень, який в свою чергу сигналізує про це оператору, який повинен проаналізувати об'єкт і виконати відповідні дії.

3. Рішення проблеми виходу з ладу датчиків. Агент системи контролю через інтервал часу Δt опитує всіх агентів датчиків на працездатність, якщо відповідь не отримана, то датчик вважається що з ладу. Агент системи контролю передає його функції сусіднім датчикам, якщо датчик, вийшов з ладу був граничним або тривожним, з метою, щоб вся територія на яку веде шлях з місця спостереження зламаного датчика перебувала під постійним наглядом. Якщо ж датчик був звичайним, то про його несправність агент системи контролю попереджає сусідні датчики. Доки його не полагодять, вони враховують похибка знаходження спостережуваних об'єктів. У разі відновлення датчика або появи нового датчика в системі реалізований алгоритм, який допомагає визначити до якого типу повинен бути віднесений агент цього датчика - граничний, тривожний або звичайний. Це досягається завдяки наявності інформації про всі датчики систему у агента системи контролю, і приймає рішення про присвоєння типу для агенту датчика.

4. Комунікації між оператором та агентами через агента підтримки прийняття рішень. Агенти повинні за запитом оператора через агента підтримки прийняття рішень передавати йому певну інформацію, а також робити це без запиту за визначеною ним схемою в певних часових проміжках (збір статистики). Оператора при цьому цікавить місце розташування об'єктів на ділянці, напрямки їх руху, працездатність всіх датчиків, поточні енерговитрати системи, близькість спостережуваних об'єктів до захищається, таких як дитина, машина, гараж, будинок. Алгоритм враховує такі команди: передати поточний стан об'єкта, передати працездатність здатність датчиків, передати поточні енерговитрати, встановити певний пріоритет для певного об'єкта, передати напрямку руху об'єкта по координатах або його тимчасовим номером. Всі ці команди реалізовані на програмному рівні і служать зручним способом зв'язку користувача (оператора-охоронця) з програмним забезпеченням розробленої системи (мультиагентної системою охорони об'єкта).

Висновки

В наслідок застосування запропонованої методики роботи мультиагентної системи на базі онтології OWL для системи охорони об'єкта досягнуто економію електроенергії шляхом перемикання окремих датчиків в сплячий режим; зменшення ціни на охоронну систему через використання дешевих датчиків; підвищення ефективності роботи за рахунок передачі функції з контролю за об'єктом, що охороняється агенту підтримки прийняття рішень; забезпечення розширення системи за допомогою використання мультиагентної системи заснованої на онтології OWL. Завдання, які реалізуються в рамках цього рішення дозволяють побудувати маршрут руху об'єктів на місцевості, передбачити їх можливий напрямку руху, дозволяє попередити злочинні дії і підвищити шанси по реєстрації особистості злочинця. Підсумком дослідження є створення конкурентної методики, здатної в багатьох аспектах переважити існуючі рішення.

Література

- [1] Бугайченко Д.Ю. Разработка и реализация методов формально-логической спецификации самонастраивающихся мультиагентных систем с временными ограничениями: дис. к-та ф-м. наук: 05.13.11 / Д.Ю. Бугайченко. - Санкт-Петербург, 2007. - 259с.
- [2] Витих В.А. Мультиагентные модели взаимодействия для построения сетей потребностей и возможностей в открытых системах / В.А. Витих, П.О. Скобелев // Автоматизация и телемеханика. - 2002. - №1. - С. 177-185.
- [3] Скобелев П.О. Открытые мультиагентные системы для оперативной обработки информации в процессах принятия решений: дис. д-ра техн. наук: 05.13.01 / П.О. Скобелев. - Самара, 2003. - 415с.
- [4] Планирование целевого применения группировки космических аппаратов дистанционного зондирования Земли с использованием мультиагентных технологий / П.О. Скобелев, В.К. Скирмунт, Е.В. Симонова [и др.] // Известия ЮФУ. Технические науки. - 2015. - №10. - С.60-70.
- [5] Применение онтологии в интеллектуальной системе распределенного управления малоразмерных космических аппаратов / П.О. Скобелев, Е.В. Симонова, М.Е. Степанова [и др.] //
- [6] Известия Самарского научного центра РАН. - 2015. - №2. - С.1119-1130.
- [7] Garcia J. Agent-based Coordination of Cameras / J. Garcia, J. Carbo, J.M. Molina // International Journal of Computer Science and Applications. - 2005. - №2. - pp. 33-37. Agent OWL: Semantic knowledge model and agent architecture / M. Laclavik, Z. Balogh, M. Babik [and etc.] // Computing and Informatics. - 2006. - №25. - pp. 419-437.
- [8] Vivek K.S. Cooperative multi-camera surveillance using model predictive control / K.S. Vivek, P.K. Atrey, M.S. Kankanhalli. - 2008. - №19. - pp. 375-393.
- [9] FIPA Abstract Architecture Specification. SC00001L. - 2002.

References

- [1] D.Y. Bugaychenko, "Razrabotka i realizaciya metodov formalno-logicheskoy specifikacii samonastravayushhixsya multiagentnyx sistem s vremennymi ogranicheniyami", Ph.D dissertation, Dept. Phys-Math, SPU, Russia, 2007.



- [2] V.A. Vittix, P.O. Skobelev “Multiagentnye modeli vzaimodejstviya dlya postroeniya setej potrebnostej i vozmozhnostej v otkrytyx sistemax”, avtomatizaciya i telemex., № 1, pp. 177-185, 2002.
- [3] P.O. Skobelev i dr. “primenenie ontologii v intellektualnoj sisteme raspredelenного upravleniya malorazmernyx kosmicheskix apparatov”, izvestiya samarskogo nauchnogo centra ran, Vol. 17, № 2(5), pp. 1119-1130, 2015.
- [4] P.O. Skobelev i dr. “Planirovanie celevogo primeneniya gruppировки kosmicheskix apparatov distancionnogo zondirovaniya zemli s ispolzovaniem multiagentnyx texnologij”, izvestiya yufu. texnicheskie nauki, №10(171), pp. 60-70, 2015.
- [5] P.O. Skobelev, “Otkrytye multiagentnye sistemy dlya operativnoj obrabotki informacii v processax prinyatiya reshenij”, Ph.D. dissertation, Dept. Tech, ICCS, Samara, Russia, 2003.
- [6] J. Garcia et al., “Agent-based Coordination of Cameras”, International Journal of Computer Science and Applications, vol. 2, pp. 33-37, Jan. 2005.
- [7] M. Laclavik et al. “Agent OWL: Semantic knowledge model and agent architecture”, Computing and Informatics, vol. 25, pp. 419-437, May. 2006.
- [8] K.S. Vivek et al., “Coopetitive multi-camera surveillance using model predictive control”, Machine Vision and Applications, vol. 19, pp. 375-393, Oct. 2008.
- [9] FIPA Abstract Architecture Specification, SC00001L, 2002.

УДК 621.86.08-52

ІМІТАЦІЙНА МАТЕМАТИЧНА МОДЕЛЬ КОНВЕЄРНИХ ВАГОВІМІРЮВАЧІВ ДОЗАТОРІВ БЕЗПЕРЕРВНОЇ ДІЇ

Д.В. Дец¹, К. Шейда-Голбад², В.А. Хобін³

^{1,2,3} Одеська національна академія харчових технологій, Україна

ORCID: ¹ 0000-0002-8556-5451, ³ 0000-0003-0238-8371

E-mail: ¹ dec@onaft.edu.ua, ² kambiz@ukr.net, ³ khobin@onaft.edu.ua

Copyright © 2017 by author and the journal “Automation technological and business - processes”.

This work is licensed under the Creative Commons Attribution International License (CC BY).

<http://creativecommons.org/licenses/by/4.0/>



DOI: [10.15673/atbp.v10i4.826](https://doi.org/10.15673/atbp.v10i4.826)

Анотація: У статті розглянуто імітаційне моделювання конвеєрних ваговимірювачів дозаторів безперервної дії у середовищі MATLAB\Simulink. Такі дозатори атестуються метрологічними службами як вимірювальні пристрої. Це вимагає точного відображення моделюємих властивостей ваговимірювачів. Отриманий комплекс імітаційних цифрових моделей процесів переміщення та зважування дозуємого сипучого матеріалу, які реалізує конвеєрний ваговимірювач, дозволить проводити коректні машинні експерименти для розробки та тестування ефективних алгоритмів керування автономними дозаторами та їхніми комплексами.

Abstract: In the article simulation modeling of conveyor weight meters of continuous batchers in MATLAB \ Simulink environment is considered. Such batchers are certified by metrological services as measuring devices. It requires an accurate reflection of the simulated properties of the weight meters. The obtained complex of imitation digital models of the processes of movement and weighing of the batched bulk material, which the conveyor weighing meters are realized, allows to conduct correct machine experiments for the development and testing of effective control algorithms by autonomous batchers and their complexes.

Ключові слова: дозатор, вагове безперервне дозування, конвеєрний ваговимірювач, імітаційна математична модель.

Keywords: feeder, weight continuous dosing, conveyor belt scales, simulation modeling.
