

УДК 621.396

О.А. Симоненко, О.Я. Сова, В.А. Романюк, Я.Л. Уманець

*Військовий інститут телекомунікацій та інформатизації  
Державного університету телекомунікацій, Київ*

## АНАЛІЗ ІСНУЮЧИХ АГЕНТНИХ ПЛАТФОРМ ДЛЯ ПОБУДОВИ СИСТЕМ УПРАВЛІННЯ ВУЗЛАМИ МОБІЛЬНИХ РАДІОМЕРЕЖ КЛАСУ MANET

*Проведено аналіз існуючих агентних платформ та можливість їх застосування для створення системи управління вузлами мобільних радіомереж класу MANET.*

**Ключові слова:** агент, агента платформа, мобільні радіомережі.

### Вступ

Сучасні системи безпроводового зв'язку розвиваються у напрямку забезпечення мобільності їх абонентів та організації зв'язку між абонентами за принципом „у будь-якому місці, у будь-який час”. Прикладом реалізації даного принципу є мобільні радіомережі (МР), що будуються за технологією MANET (*Mobile Ad-Hoc Networks*) [2]. Особливості функціонування МР вимагають наявності в складі кожного вузла системи управління (СУ) [1], здатної приймати рішення в умовах невизначеності. В ході попередніх досліджень показано, що реалізація підсистем вузлової СУ можлива шляхом використання агентів.

Під агентом розуміється деяка автономна програма, яка знаходиться в середовищі, від якого отримуються дані про події, що відбуваються, інтерпретує їх і виконує команди, що впливають на це середовище, в залежності від поставленої мети.

Зважаючи на особливості функціонування СУ вузлами МР, які, з одного боку передбачають відносно автономність при вирішенні завдань з управління вузловими ресурсами, а з іншого боку – повинні враховувати стан вузлів-сусідів при прийнятті управлінських рішень (поодинокі вузлові СУ не зможуть вирішити всіх завдань поставлених перед ними), запропоновано для побудови структурних елементів СУ вузлами радіомереж класу MANET використовувати технологію інтелектуальних агентів, з можливістю їх об'єднання в багатоагентну систему, яка працює „поверх” агентної платформи (АП) і використовує її сервіси.

### Агентні платформи

Базовим інструментом розробки інтелектуальних багатоагентних систем, що дозволяє створювати, знищувати, інтерпретувати, запускати і переміщати агентів є агентна платформа. Агентна платформа – це програмне середовище, що реалізує основні механізми, які забезпечують роботу багатоагентних систем і полегшує розробку агентних систем та

агентів взагалі. Основними функціями агентних платформ є:

- організація взаємодії агентів;
- передача повідомлень між агентами всередині платформи (на різних рівнях: на рівні мережевих пакетів, повідомлень будь-якої мови реалізації, протоколів передачі) і між різними платформами;
- підтримка онтологій;
- управління агентами, їх життєвими циклами;
- пошук агентів і даних про них всередині системи;
- забезпечення безпеки агентів.

Найбільш відомі такі агентні платформи [5]:

- JADE (Java Agent Development Framework);
- Coguaar (Cognitive Agent Architecture) – найпотужніший проект, підтримується DARPA (Defense Advanced Research Projects Agency – агентство передових оборонних дослідницьких проектів);
- Aglobe – володіє недостатньою підтримкою FIPA (Foundation for Intelligent Physical Agents) – організації стандарту IEEE, яка сприяє розвитку технології на основі агентів і взаємодії їх з іншими технологіями;

– Jason (multi-agent systems development platform) – для розробки багатоагентних систем, має багато функцій, що можна налаштувати;

– Jack – одна з небагатьох платформ, де використовуються модель логіки агента, що заснована на принципах переконання-бажання-наміри (Belief-desire-intention software model – BDI) і має вбудовані формально-логічні планування роботи агента.

Крім агентних платформ для програмування агентів можуть застосовуватися:

- універсальні мови (Java, C++, Visual Basic, C#);
- мови представлення знань (SL, KIF);
- мови комунікацій та обміну знаннями (KQML, AgentSpeak, April);
- мови сценаріїв (TCL/TK, Python, Perl 5);
- спеціалізовані мови (TeleScript, COOL, Agent0, AgentK);

– символічні мови та мови логічного програмування (Oz, ConGolog, IMPACT, Dylog, Concurrent METATEM);

– а також інші мови і засоби розробки агентів.

Проведемо короткий аналіз наведених вище платформ з метою визначення можливості їх використання для розробки агентів систем управління вузлами МР класу MANET.

**Платформа JADE.** До основних характеристик агентної платформи JADE належать [6]:

- відповідність останнім специфікаціям FIPA 2.0;
- відкритість коду LGPL (Lesser General Public License);
- підтримка мультиплатформних технологій;
- підтримка бездротових мобільних пристроїв аж до JME (Java Platform, Micro Edition) MIDP (Mobile information device profile) 1.0 (стільникові телефони);
- наявність власних і сторонніх бібліотек розширення протоколів взаємодії агентів (RDF, XML);
- управління агентами;
- управління самою платформою;
- статистика.

На базі даної агентної платформи можливо реалізувати автономного агента, який зможе повністю контролювати свої управляючі впливи в межах покладених на нього функцій, самостійно управляти своїм життєвим циклом, зможе виконувати декілька паралельних завдань. Так як JADE підтримує внутрішньо-платформенні мобільність і клонування агентів, то на практиці платформа може бути розподілена між кількома вузлами МР. Кожен вузол може мати у своєму складі декілька агентів, кожний з яких відповідатиме за виконання своєї задачі і, разом з тим, зможе переміщатися з одного вузла МР до іншого у разі необхідності. До недоліків даної агентної платформи можна віднести відкритість коду, що недопустимо при розробці засобів зв'язку спеціального призначення.

**Платформа Cougaar.** Розглядаючи агентну платформу Cougaar можна відзначити, що вона має архітектуру з відкритим кодом. Агенти є автономними утвореннями програмного забезпечення, які взаємодіють з іншими агентами, так і з зовнішнім середовищем, що зможе забезпечувати надійне функціонування вузла. Програмні агенти реалізуються на основі методології програмування, що дозволяє розкладати складні завдання на більш прості. Агенти матимуть можливість керувати вузлом МР, змінювати його стан відповідно до змін навколишнього середовища. Cougaar включає в себе кілька сучасних комплексних можливостей:

- зв'язок всередині самого агента і між агентами;
- сервіси, що засновані на представленні знань системи управління;
- механізми, які дозволяють реалізувати координацію між агентами.

Агент, що функціонуватиме на вузлі (технологія *Java Virtual Machine*), самостійно узгоджуватиме роботу з іншими агентами, а також зможе визначати поведінку всієї системи управління вузлом МР. Всі правила зберігатимуться в базі знань, що забезпечуватиме відновлення системи після збоїв, та відповідну реакцію при змінах в навколишньому середовищі, або обстановки на полі бою. До недоліків даної платформи також можна віднести відкритість програмного коду.

**Платформа Aglobe** є агентною платформою, що призначена для тестування експериментальних сценаріїв за участю агентів, тестування їхнього стану і зв'язків між декількома агентами. Платформа реалізує функції для агентів, такі як комунікаційна інфраструктура, взаємодія агентів, розгортання агентів. *Aglobe* – платформа, що несумісна зі специфікаціями FIPA. Вона не підтримує взаємодію з іншими агентними платформами (при розробці закритих систем така взаємодія непотрібна).

Для реалізації великих систем, що потребують значних вимог до ресурсів, швидкості передачі даних, дана платформа є досить слабкою. *Aglobe* підходить для реального моделювання статичних об'єктів (стаціонарні вузли, пункти управління) і мобільних пристроїв (рухомі вузли). У такому випадку платформа може бути запущена в розширеному варіанті з підтримкою GIS (*Geographical Information System*) та симулятора навколишнього середовища ES (*Environment Simulator*). ES-агент зможе імітувати динаміку (фізичне місце розташування, рух у часі та інші параметри) кожного вузла.

Перевагами даної платформи є можливість налагодження тестових агентів, та взаємозв'язків між агентами. А основним недоліком є повна несумісність з іншими платформами.

**Платформа Jason** є платформою для розробки багатоагентних систем, розширення агент-орієнтованої мови програмування *AgentSpeak* використовується для програмування поведінки окремих агентів. *Jason* розроблений в *Java* і дозволяє настраювати більшість аспектів агента або мультиагента. Він реалізується у вигляді *plug-in* (незалежно компільований програмний модуль, що динамічно підключається до основної програми, призначений для розширення або використання її можливостей) для одного програмного агента (ПА), а також для розгортання багатоагентних систем, або в якості агента на основі розподіленої системи проміжного програмного забезпечення. *Jason* володіє відкритим вихідним кодом і розповсюджується під ліцензією GNU (*General Public License*) LGPL.

*Jason* має основні можливості:

- сильний інструментарій для опису навколишнього середовища;
- обробка збоїв, та відповідна реакція на них;

- взаємодія між агентами, перевірка достовірності агентів;
- підтримка багатоагентних ситем;
- можливість реалізації багатоагентної системи, що розподілена між вузлами мережі;
- налаштування функцій та загальної архітектури;
- наявність бази знань.

До недоліків даної платформи також можливо віднести відкритість програмного коду. Перевагами ж даної платформи є потужний інструментарій властивостей.

**Платформа Jack** – платформа для реалізації інтелектуальних агентів або багатоагентних систем, що використовує орієнтоване програмне забезпечення і є третім поколінням платформи агента, що застосовує технології PRS (*Procedural Reasoning System*) та dMARS (*Distributed Multi-Agent Reasoning System*). Jack є однією з небагатьох багатоагентних систем, що використовує модель BDI програмного забезпечення і надає свою власну реалізацію за допомогою мови

програмування *Java*, а також графічні інструменти планування. Платформа *Jack* складаються з набору автономних агентів, що можуть отримувати інформацію з навколишнього середовища і взаємодіяти з іншими агентами. Кожен агент матиме свою мету, базу знань для виконання покладених на нього функцій. Це дуже ефективний спосіб створення додатків для динамічних і складних систем – кожен агент несе відповідальність за власні цілі, реагуючи на події та поведінку інших агентів в системі. Немає необхідності явної взаємодії агентів всієї системи, зв'язки виникають як наслідок індивідуальних цілей.

*Jack* – середовище для побудови, експлуатації та інтеграції багатоагентних систем. Він також використовує технологію BDI. Дана платформа може працювати з декількома потоками між декількома вузлами і легко взаємодіє із сторонніми бібліотеками. Дана платформа має структуру, яка найбільше відповідає для реалізації вузлової СУ з використанням технології агентів відповідно до специфікації FIPA (рис. 1):

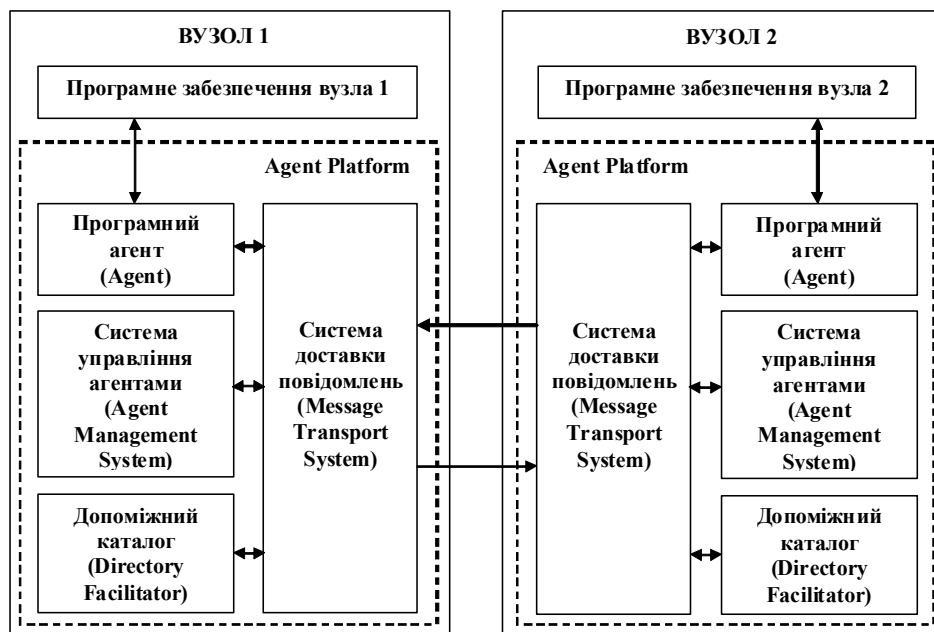


Рис. 1. Структурна схема взаємодії агентних платформ згідно рекомендацій FIPA (Foundation for Intelligent Physical Agents)

– Agent – це програма, яка знаходиться в середовищі, від якого отримуються дані про події, що відбуваються, інтерпретує їх і виконує команди, що впливають на це середовище [7];

– Agent Management System (AMS) – обов'язковий компонент АП, де мають реєструватися агенти;

– Message Transport System (MTS) – забезпечує комунікацію між агентами на різних агентних платформах;

– Agent Platform (AP) – фізична інфраструктура, у якій функціонують агенти;

– Directory Facilitator (DF) – додатковий, не обов'язковий компонент, що забезпечує довідкову

інформацію іншим агентам. Тут агенти можуть реєструвати свої послуги або здійснювати запити, щоб довідатися про наявні можливості інших агентів.

Основний принцип архітектури полягає в тому, щоб представити сукупність агентів як багаторівневу структуру з блоком управління, що використовує загальну базу знань.

### Можливість застосування АП для побудови СУ вузлами МР

Розглянуті вище АП дають можливість розробляти багатоагентні системи для побудови СУ вузлами радіомереж класу MANET [3]. В якості критеріїв вибору засобів розробки можна використовувати, табл. 1.

Таблиця 1

Порівняння засобів  
реалізації агентів за критеріями

Засоби розробки агента/ мова реалізації	Критерії						
	a	b	c	d	e	f	g
<i>Mscrosoft.NET/C#</i>	+	+	-	+	+	+	+
<i>NetStepper Pro 1.0/Java</i>	+	+	-	+	+	+	+
<i>JADE/Java</i>	+	+	-	+	+	+	+
<i>Aglobe/Java</i>	+	+	-	+	+	+	+
<i>Jason/Java</i>	+	+	-	+	+	+	+
<i>Jack/Java</i>	+	+	+	+	-	+	-

- а-критерій: здатність до самоорганізації;
- б-критерій: здатність до функціонування в режимі реального часу;
- с-критерій: наявність правил поведінки та моделей реакцій на основі даних, отриманих з зовнішнього середовища;
- d-критерій: активізації інших агентів у разі зміни параметрів зовнішнього середовища чи після надходження команд з СУ;
- е-критерій: забезпечення самонавчання СУ на основі зібраної інформації про стан навколишнього середовища та стан самої системи управління;
- f-критерій: визначення (зміна, у разі необхідності) пріоритетів функціонування агентів в залежності від доступних йому внутрішніх ресурсів, що дозволить уникнути ситуації, за якої прийняття управляючих рішень буде покладене на агента з недостатніми внутрішніми ресурсами;
- g-критерій: виконання функцій тимчасового координатора МР, при втраті зв'язку з вузлом, який координує роботу МР, в залежності від визначеної в процесі функціонування пріоритетності.

## Висновки

Таким чином, при застосуванні агентів в МР класу MANET [7], задоволення зазначених вище критеріїв є важливою умовою для їх успішної реалізації з використанням того чи іншого програмного

засобу чи платформи. Як видно з таблиці, цим критеріям задовольняють багато програмних засобів реалізації агентів, однак переважна більшість платформ використовує мову програмування *Java*, як таку, що дозволяє реалізувати агенти, задовольняючи всі вище зазначені критерії. Для реалізації вузлової СУ МР доцільно використовувати агентну платформу *Jack*, адже вона може мати у своєму складі автономні агенти, в яких наявна власна база знань, що дозволить розробляти інтелектуальні агенти [4] та створити інтелектуальну систему управління МР.

## Список літератури

1. Романюк В.А. Інтелектуальні мобільні радіомережі / В.А. Романюк // *V НТК „Пріоритетні напрямки розвитку телекомунікаційних систем та мереж спеціального призначення”: матеріали конференції (Київ, 20–21 жовт. 2010 р.)*. К.: ВІТІ НТУУ „КПІ”, 2010. – С. 28 – 36;
2. *Broadcasting in Multi-Radio Multi-Channel Wireless Networks using Simplicial Complexes/ Ren W., Zhao Q., Ramanathan R., Gao J., Swami A., Bar-Noy A., Johnson M., Basu P. // Mobile Adhoc and Sensor Systems (MASS), 2011 IEEE 8th International Conference, On page(s): 660 – 665;*
3. Концепция иерархического построения интеллектуальных систем управления тактическими радиосетями класса MANET / В.А. Романюк, О.Я. Сова, П.В. Жук, А.В. Романюк // *Сб. тез. докл. XXII Межд. Крымской конференции [“СВЧ-техника и телекоммуникационные технологии”]*, (КрыМуКо). – Севастополь, 2012. – С. 265;
4. Karthik D. Situation based intelligence routing in wireless sensor network / D. karthik, S. Nagarajan // *International Conference on Computational Intelligence and Computing Research – 2011;*
5. Амелин К.С. Применение мультиагентных технологий для управления группой легких мобильных роботов / К.С. Амелин, Н.О. Амелина // *АТМОСФЕРА: Встраиваемые системы и робототехника – 2012;*
6. F.L. Bellifemine, G. Caire, and D. Greenwood, *Developing Multi-Agent Systems with JADE*. Wiley, 2007.
7. Аналіз можливостей використання інтелектуальних агентів для побудови системи управління вузлами радіомереж класу MANET // О.А. Симоненко, Я.Л. Уманець, В.А. Романюк, О.Я. Сова // *Збірник наукових праць ВІТІ НТУУ “КПІ”*. – 2013.

Надійшла до редколегії 4.12.2013

**Рецензент:** д-р техн. наук, проф. О.В. Кувшинов, Військовий інститут телекомунікацій та інформатизації Державного університету телекомунікацій, Київ.

## АНАЛИЗ СУЩЕСТВУЮЩИХ АГЕНТНЫХ ПЛАТФОРМ ДЛЯ ПОСТРОЕНИЯ СИСТЕМ УПРАВЛЕНИЯ УЗЛАМИ МОБИЛЬНЫХ РАДИОСЕТЕЙ КЛАССА MANET

А.А. Симоненко, О.Я. Сова, В.А. Романюк, Я.Л. Уманець

*Проведен анализ существующих агентных платформ и возможность их применения для создания системы управления узлами мобильных радиосетей класса MANET.*

**Ключевые слова:** агент, агентная платформа, мобильные радиосети.

## ANALYSIS OF THE EXISTING AGENT PLATFORMS FOR BUILDING MANAGEMENT SYSTEMS MOBILE RADIO NODES CLASS MANET

O.A. Simonenko, O.Y. Sova, V.A. Romanyuk, Y.L. Umanets

*Existing agent platforms and their applicability to the MANET-nodes control system are analyzed.*

**Keywords:** agent, platform's agent, mobile radio networks.