КІБЕРФІЗИЧНІ СИСТЕМИ ТА ЇХ ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ

CYBER-PHYSICAL SYSTEMS AND THEIR SOFTWARE

Ван Чунжі[1], д-р техн. наук, проф., Яцишин С. П.[2], д-р техн. наук, проф.,
Лиса О. В.[3], канд. техн. наук, доц., Мідик А-В. В.[4], аспірант
[1]School of Computer Science, Hubei University of Technology, China;
[2]Національний університет «Львівська політехніка», Україна;
[3]Львівський аграрний університет, Україна
[4]Національний університет «Львівська політехніка», аспірантура, Україна;
e-mail: ovl2407@ukr.net

https://doi.org/10.23939/istcmtm2018.01.034

Анотація. У зв'язку зі швидким розвитком кіберфізичних систем істотну увагу в світі звертають на різні аспекти їх формування й експлуатації. У 2014 р. NIST, USA створила Громадську робочу групу кіберфізичних систем для об'єднання широкого кола фахівців на відкритому громадському форумі, щоб допомогти визначити та сформувати їхні основні характеристики для керівництва розробленням та впровадженням «інтелектуальних» програм у різних сферах, серед яких розумні виробництво, транспортування, енергетика та охорона здоров'я. Важливим вважається питання програмного забезпечення, оскільки, по-перше, вказані системи складаються із великої, нерегламентованої кількості компонентів, рознесених у просторі й часі, а, по-друге, компоненти систем мають змогу самостійно, під власні потреби, додатково встановлювати необхідне програмне забезпечення, відоме як Middleware.

Ключові слова: кіберфізичні системи, смарт-засоби, програмне забезпечення, проміжне програмне забезпечення.

Annotation. In connection with the enhanced development of cyber-physical systems, considerable attention in the world is given to various aspects of their formation and operation. So in 2014, NIST, USA has created a Cyber-Physical System (further CPS) Community Working Group to bring together a wide range of professionals at an open, public forum to help identify and shape the main characteristics of the CPS to guide the design and implementation of «intelligent» programs in various areas, including smart production, transportation, energy and healthcare.

By analyzing the known CPSs, their metrological and software, we can evidence that their number is steadily increasing, and the scope of application is expanding. Software and metrological assurance develop in the direction of supporting the work of existing CPSs as also and in the design of their suitability for new types of CPSs. Therefore, the requirements for creating a CPS are security, confidentiality, reliability, stability, guarantees for common interconnected devices and infrastructures, dynamism, compatibility (the ability to host different computing models), the support of different modes of communication in the network, the solvability of complexity problems (problems of accessing the obtained data and control with feedback in any architecture of the CPS), synchronization, interaction with the operating environment, the ability to cooperate with each other for the creation of the complex effects, the creation of effects that exceeds the sum of the effects of individual parts of the CPS, the ability to combine several goals.

CPSs are characterized by well-defined components: with known performance described by using standardized semantics and syntax. CPSs must support the flexibility of applications and domains. To realize this, the definition of components must be flexible and open. The architecture should support an accurate questioning of components in order to provide flexibility in the creation and adaptation of virtual systems and the promotion of innovation. The CPS should support a large scale of sizes, complexity and loading in addition. The components should be integrated or/and scaled quickly, even while operating. The CPS architecture should consist of independent, disconnected components for flexibility, reliability and resilience to changing situations. The solution must also exist between the architectural layers, allowing each layer to be changed, without affecting other layers. In order for the system to integrate different components, the interfaces to these components should be based on interpreted and unambiguous standards. Adaptation is achieved through the flexibility of internal components and interoperability. An important issue is software, since, firstly, these systems consist of a large, unregulated number of components spaced apart in space and time, and secondly, system components are able to independently install themselves for their own needs the required software denominated as Middleware.

Key words: cyber-physical systems, smart means, software, middleware.

Вступ

Кіберфізичні системи (далі – КФС) – це інтелектуальні системи, у які входять мережі фізичних та обчислювальних компонентів, що інженерно взаємодіють. Вони проникають у всі сфери життєдіяльності людини: виробництво, будівництво, транспорт, енергетику, медицину тощо, де забезпечують нові функціональні можливості для покращення якості життя, досягнення технічного прогресу в різних сферах і тому істотно впливають на світову економіку. Основою розроблення різних моделей кіберфізичних систем є наявність засобів вимірювання та їх програмного забезпечення. Засоби необхідні для контролю параметрів технологічних процесів та навколишнього середовища [1].

Суть «інтелектуальних» програм полягає в тому, що, використовуючи дані сенсорів, які якнайшвидше і якнайточніше сигналізують про зміну параметрів середовища, спеціальні алгоритми задіюють автоматику вищого рівня для виконання адекватних дій. КФС виходить за межі звичайного продукту, системи та архітектури прикладних програм. Зазвичай КФС охоплює всі відомі аспекти роботи інформаційно-вимірювальних систем, ускладнених унаслідок взаємодії їхніх окре-

мих компонентів через мережі. Вони об'єднують традиційні інформаційні технології: від надходження даних від сенсорів з їх опрацюванням із використанням вбудованих обчислювальних потужностей або з використанням хмарних технологій, до традиційних операційних технологій контролю та управління. Інакше кажучи, особливістю КФС є поєднання інформаційних та операційних технологій, на що накладаються часо-просторові обмеження, оскільки КФС часто розпорошені у просторі та розділені у часі.

Питання упровадження «інтелектуальних» програм у різних сферах настільки актуальне, що NIST розробила класифікацію КФС, яка охоплює розумне виробництво (пряме та додаткове), розумні конструкції, розумний транспорт, розумну енергетику, розумну безпеку життя та розумну охорону здоров'я. Щоб забезпечити впровадження «інтелектуальних» програм, необхідно створити прикладну модель системи, мати відповідне метрологічне та програмне забезпечення. Крім того, треба добитись сумісності між різнорідними компонентами та системами, тому потрібні розробки у сфері метрології (калібрування, оцінювання якості комплексних продуктів, діагностика на основі моделі), у сфері розроблення основного і проміжного програмного забезпечення. Програмне забезпечення формує адекватну прогнозовану поведінку системи – відповідь системи на зміну згаданих параметрів.

Недоліки

На особливу увагу заслуговує режим роботи КФС без участі фахівців (англ. – «*man-out-loop*» regime), який у поєднанні із гнучкістю системи, забезпечуваною автоматичним оновленням програмного забезпечення та впровадженням проміжного програмного забезпечення (англ. – *Middleware*), може призвести до виникнення критичних ситуацій у працездатності й ефективності роботи систем загалом. Наприклад, метрологічна відмова, спричинена погіршенням параметрів систем медичного догляду (автоматичного вприскування ліків із інстальованої під шкірою пацієнта капсули), може призвести до незворотних наслідків.

Мета роботи

Метою роботи є дослідження працездатності й ефективності кіберфізичних систем у результаті аналізу особливостей їх метрологічного та програмного забезпечення.

Матеріали та методи

Аналізуючи відомі сьогодні КФС, їх метрологічне та програмне забезпечення, бачимо, що невпинно зростає їх кількість, розширюються сфери застосування. Програмне та метрологічне забезпечення розвивається в напрямі підтримання роботи наявних КФС та їх придатності для конструювання нових КФС. Тому вимогами, які ставлять, створюючи КФС, є безпека, конфіденційність, надійність, стійкість, гарантії щодо поширених взаємопов'язаних пристроїв та інфраструктур, динамічність, сумісність (можливість розмістити різні обчислювальні моделі), підтримуваність різних режимів спілкування у мережі, вирішуваність проблем складності (проблеми зондування та керування зі зворотним зв'язком у будь-якій архітектурі КФС), синхронізація, взаємодія із середовищем експлуатації, можливість різних КФС співпрацювати для створення ефектів, більших за суму частин окремих КФС, можливість поєднання декількох цілей.

КФС характеризуються чітко визначеними компонентами: з відомими характеристиками, описаними з використанням стандартизованої семантики та синтаксису. КФС повинні підтримувати гнучкість додатків та доменів. Для цього визначення компонентів повинно бути гнучким і відкритим, архітектура повинна підтримувати точне опитування смарт-речей, щоб забезпечити гнучкість у створенні та адаптації віртуальних систем, та сприяння інноваціям. КФС повинні підтримувати великий діапазон розмірів, складності та навантаження на додаток. І у простій, і у складній розподіленій системі необхідно використовуватися ті самі компоненти. Компоненти мають бути зібрані та масштабовані швидко, навіть під час роботи. КФС повинна складатися з незалежних компонентів для забезпечення гнучкості, надійності та стійкості до змінних ситуацій. Розв'язки повинні існувати між архітектурними шарами, що дає змогу змінювати кожен шар, не впливаючи на інші шари. Для того, щоб система могла інтегрувати різні компоненти, інтерфейси до цих компонентів повинні основуватись на інтерпретованих та однозначних стандартах. Стандартизація інтерфейсів дасть змогу забезпечити різні компоненти наявних і майбутніх систем. Адаптація досягається через гнучкість внутрішніх компонентів та зовнішню сумісність.

Інакше кажучи, до КФС ставлять різноманітні вимоги. Тому вважаємо за доцільне охарактеризувати кожен з видів КФС з урахуванням метрологічного та програмного забезпечення, яке використовують під час їх конструювання.

1. Види КФС

1.1. КФС «Розумне виробництво» – багатофункціональні смарт-машини, вирізняються малими розмірами, адаптивністю до потреб користувачів (реалізується збиранням потрібної функціональності на одній машині). Отримавши інформацію про

змінені вимоги, КФС сама вносить корективи в технологічний процес. Прикладом розумного виробництва є виробництво металу з використанням точних ваг. Їх нормальне функціонування забезпечується калібруванням, виконуваним на місці, без демонтажу конструкції КФС. До прикладу, завдяки «man-in-loop» технології здійснюється дистанційна атестація точних ваг, що сприяє підвищенню їхньої продуктивності в певних операційних діапазонах до значень робочих еталонів. Переважно для таких КФС застосовують програмну оболонку Linux, а віддалений доступ до вагопроцесорів компонентів КФС безпосередньо на робочих місцях забезпечують завдяки використанню промислового Ethernet як стандартизованого варіанта мережевого протоколу Ethernet, адаптованого для промислових умов, з метою автоматизації та керування технологічними процесами.

1.2. КФС «Розумні будівлі» – інтелектуальні будівлі (з мінімальним чи нульовим споживанням ресурсів), що потребують постійного моніторингу, вони повинні бути підключені до мереж інтелектуальних сенсорів і контролюватися засобами КФС. Основною вимогою є досягнення нульового споживання енергії. Для цього вивчають теплові умови за допомогою інтелектуальних сенсорів температури локальної мережі та забезпечують адекватну ізоляцію за безперервного багатоточкового контролю температури. Велику увагу приділяють попередній оцінці теплових умов на етапі будівництва, усуненню містків холоду тощо. Для цього використовують такі методи і засоби контролю тепла: методи інфрачервоної діагностики за допомогою тепловізора, метод дискретно-точкового вивчення температурного режиму обмежувальних площин, моніторинг температур у часі за допомогою чипів із вбудованими сенсорами температури, прикріпленими до внутрішньої та зовнішньої поверхонь обмежувальних площин тощо. Так виявляють та усувають містки холоду, досліджують енергетичні відбивні покриття з невідомим коефіцієнтом чорноти і керують роботою енергетичних підсистем для електропостачання, опалення та вентиляції. У переважній більшості таких КФС застосовують програмне забезпечення StructureWare Building Operation.

Щоб поведінка програмного забезпечення КФС відповідала його специфікації під час використання в певному середовищі, проводять функціональну перевірку, орієнтуючись на детерміністичні й стохастичні системи. Специфікація виражає стан безпеки, якої треба досягти за всіх можливих виконань програмного забезпечення (для детерміністичних систем) або з необхідною мінімальною вірогідністю (для стохастичних систем). КФС працюють у недетермінованих середовищах. Стосовно програмного забезпечення це означає, що деякі його входи – випадкові величини. Тому обчислюють ймовірність того, що програмне забезпечення відповідає специфікаціям безпеки. Відомі два методи перевірки КФС: імовірнісна модель перевірки (система моделюється як ланцюжок Маркова, а ймовірність обчислюють за допомогою побудови множини рівнянь та їх числового розв'язання), статистична перевірка моделі (кожне виконання системи розглядається як випробування Бернуллі, а вірогідність обчислюється за методом Монте-Карло).

1.2. Проміжне програмне забезпечення (ППЗ – Middleware) є передумовою ефективної роботи КФС [2]. Воно надає послуги програмним додаткам, окрім тих, що доступні в операційній системі, з'єднує компоненти програмного забезпечення та корпоративні програми. У ППЗ входять вебсервери, сервери додатків, системи управління контентом та аналогічні інструменти, що підтримують розроблення та доставлення додатків і уможливлюють зв'язок та керування даними в розподілених додатках. Це особливо стосується інформаційних технологій, основаних: на розширеній мові розмітки (XML); на протоколі обміну структурованими повідомленнями в розподілених обчислювальних системах під час доступу до об'єктів (SOAP); на веб-сервісах; на модульному підході (SOA) до розроблення програмного забезпечення, що оснований на використанні розподілених, слабко пов'язаних замінних компонентів, оснащених стандартизованими інтерфейсами для взаємодії за стандартизованими протоколами; на інфраструктурі Web 2.0 та на протоколі легкого доступу до каталогів (LDAP). До послуг, які можна розглядати як проміжні програми, належить інтеграція корпоративних додатків, інтеграція даних, орієнтованих на повідомлення ППЗ (MOM), брокери запиту об'єктів (ORBs) та корпоративні сервісні шини (ESB).

Іншими прикладами проміжного програмного забезпечення є:

• Програмне забезпечення Mer, яке не має ядра Linux і якому бракує інтерфейсу. Mer більше стосується мобільно-орієнтованих операційних систем постачальників обладнання.

• Операційна система Android, що використовує ядро Linux і забезпечує систему додатків, яку розробники вводять у свої програми. Крім того, Android забезпечує рівень ППЗ, зокрема бібліотеки, які надають такі послуги, як зберігання даних, екранне відображення, мультимедіа та вебпереглядач. Оскільки бібліотеки складають машинною мовою, команди виконують швидко. Бібліотеки Middleware також реалізують функції, специфічні для пристроїв, тому додатки не потре-

бують залежності від варіантів між різними пристроями Android. Проміжний шар Android також містить віртуальну машину Dalvik та її основні бібліотеки для додатків Java.

• У технології моделювання ППЗ використовується в контексті архітектури високого рівня, яка застосовується для багатьох розподілених моделювань. Це – шар програмного забезпечення, розміщений між кодом додатка та інфраструктурою під час виконання. ППЗ складається з бібліотеки функцій та дає змогу виконувати низку програм-моделювань (федератів) – пересилати ці функції із загальної бібліотеки, а не створювати їх повторно для кожної програми.

• Розробники бездротових мереж використовують ППЗ для вирішення проблем, пов'язаних із бездротовими мережами сенсора. Впровадження програми ППЗ дає змогу розробникам цих мереж інтегрувати операційні системи та апаратні засоби з різними додатками.

• Universal Home API або UHAPI – це інтерфейс прикладного програмування для приладів побутової електроніки, створений Форумом UHAPI. Мета UHAPI – дати змогу стандартному ППЗ працювати на платформах потокового аудіо/відео через апаратнонезалежний промисловий стандартний API.

• Набори програмного забезпечення для ідентифікації радіочастот забезпечують ППЗ для фільтрування зашумлених та надлишкових вихідних даних.

• ILAND – це проміжне програмне забезпечення, призначене для використання у режимі реального часу, що пропонує детерміновану підтримку реконфігурації в обмежений час.

1.3. Розумні вимірювальні засоби є важливою передумовою для створення КФС, оскільки вони являють собою основні компоненти інформаційно-вимірювальних підсистем. Розумні вимірювальні засоби поділяються на такі підкласи: розумні сенсори, розумні перетворювачі, їх мережі, які можуть спільно об'єднуватись у сучасних бездротових сенсорних мережах (БСМ).

Розумний сенсор містить в одному корпусі чутливий елемент, мікросхеми аналогового інтерфейсу, АЦП й інтерфейс шини. Крім цього, у розумному сенсорі є мікропроцесор, який кондиціює сигнали (встановлює амплітуди та форми сигналів перед відправленням їх до подальших вузлів, відфільтровує небажані шуми та виправляє помилки перед відправленням даних). Проте створення класу розумних сенсорів нового покоління передбачає необхідність у додаткових функціональних можливостях: самотестування, самоідентифікації, самовалідизації або самоадаптації. Доцільні такі можливості розумних сенсорів, як самокалібрування та самодіагностика, можливість опрацьовувати сигнал, а також мультисенсорні можливості. Конструктивно розумний сенсор температури виконується як первинний аналоговий або цифровий термочутливий перетворювач, оснащений блоком опрацювання сигналу та інтерфейсом, який може виконати низку розумних метрологічних функцій завдяки спеціальному метрологічному програмному забезпеченню.

Практично це може бути інтелектуальний сенсор температури з низкою спеціалізованих алгоритмів, передбачених на етапі проектування й виготовлення або установлених пізніше, тобто сенсор з такими вбудованими алгоритмами, які необхідні для забезпечення виконання спеціалізованих метрологічних функцій. Зокрема, такі функції передбачають здатність реалізувати автоматичне перемикання діапазонів вимірювання, залежно від значення вхідного сигналу; автоматичну самовалідизацію, самоперевірку, самодіагностику тощо; введення поправок у покази, коли зафіксовано дію фактора впливу; лінеаризацію номінальних характеристик; компенсацію температури холодного злюту термопар.

Результати й обговорення

Зважаючи на прикладні аспекти роботи КФС з їх підсистемами температурного контролю, необхідно розробляти засоби вимірювань, зокрема, разових застосувань, так звані ad-hoc підсистеми, або ж підсистеми «на вимогу». Мережа «на вимогу» є перескоковою, оскільки не стосується наперед усталеної структури, зокрема відсутні маршрутизатори у дротових мережах чи точки доступу в організованій структурі БСМ [3]. Кожен вузол мережі бере участь у маршрутизації, перекидаючи дані на сусідні вузли, оскільки визначення того, на які вузли здійснюється перекидання, виконується динамічно на основі мапи з'єднань мереж. На додаток до класичної маршрутизації мережі «на вимогу» може використовуватися фладинг (англ. Flooding) для перекидання даних. Таким є простий роутинговий алгоритм, за яким кожний вхідний пакет висилається на всі можливі адреси, за винятком того пункту, звідки надійшов. Фладинг поширений у мостах та в системах, таких як UseNet і файлообмінні мережі (peer-to-peer file sharing), а також як частина деяких роутингових протоколів, зокрема OSPF, DVMRP, а їх вже застосовують у бездротових мережах «на вимогу».

Децентралізований тип БСМ – це багатоланкові мережі, що складаються із бездротових автономних хостів. Кожен хост може слугувати маршрутизатором для передавання трафіку від інших вузлів. Бездротові однорангові мережі «на вимогу» охоплені широким спектром мережевих видозмін, що

містять сенсори, мобільний механізм перескоків, персональну специфіку чи інші мережі. Наукові розробки у цій галузі повинні передбачати навчання сенсора (переведення його з розряду інтелектуальних до розумних), вивчення безпекових аспектів за рахунок інтелектуальних можливостей вузла, зони покриття сенсора за його випадкового чи детермінованого розміщення, розташування об'єкта, визначення впливу розміщення сенсора, вивчення енергетичної ефективності передавання інформації та розкладу діяльності, маршрутизації, топологічних зв'язків, поширення даних і їх нагромадження, дослідження впливу дерева реконфігурації та конструювання конкретної топології. Для нормальної роботи мережева топологія передбачає розгляд не тільки сенсорних вузлів, а й базових станцій і крос-шарів. Структура крос-шарів важлива для БСМ, оскільки їх можна використовувати для оптимальної модуляції з метою підвищення параметрів передавання, таких як швидкість передавання даних, ефективність використання енергії тощо.

Висновки

Безперебійна та надійна робота кіберфізичних систем основана на коректній організації вхідних потоків інформації, завдяки достовірній роботі підсистем, зокрема температурного контролю, а також надійного програмного забезпечення. Наявність різних видів сенсорів для кіберфізичних систем вимагає не стільки уніфікованих вихідних сигналів різних типів сенсорів (для цього здійснюється кондиціювання їх сигналів [4]), скільки опрацювання значної, переважно надлишкової, кількості даних. Для цього треба застосовувати спеціальні алгоритми [5]. Водночас для забезпечення метрологічної достовірності за допомогою інших алгоритмів спільно опрацьовують дані, отримувані паралельно від різних типів сенсорів (data fusion). Це потребує розроблення спеціального програмного забезпечення, а воно, своєю чергою, – окремо виділеної метрологічної перевірки [6].

Подяка

Автори висловлюють вдячність колективу кафедри інформаційно-вимірювальних технологій Національного університету «Львівська політехніка» за надану допомогу та всебічне сприяння у підготовці статті.

Список літератури

1. Микийчук М. М., Стадник Б. І., Яцишин С. П., Луцик Я. Т. (2017). Розумні вимірювальні засоби для кіберфізичних систем. Вимірювальна техніка та метрологія. – № 77. – С. 3–17.

2. "What is Middleware?". Middleware.org. Defining Technology. (2008). Retrieved 2013-08-11.

3. Ad-hoc Sensor Networks. http://www.brunel.ac.uk/cedps/electronic-computer-engineering/research-activities/wncc/ad-hoc-sensor-networks

4. Дорожовець М. М. (2014). Розділ «Кондиціювання сигналів сенсорів» // у кн. «Сенсори». – Львів: Бескид Біт. – С. 124–152.

5. Яцишин С. П., Микитин І. П., Кравець І. П. (2010). Пожежні сповіщувачі. Засади оптимізації роботи та алгоритми прийняття рішень // Пожежна безпека: № 17. – С. 14–19.

6. Олеськів О. М., Микитин І .П. (2014). Аналітичний огляд процедур та методів метрологічної перевірки програмного забезпечення // Вимірювальна техніка та метрологія: № 75. – С. 2–7.

References

1. Mykyychuk M. M., Stadnyk B. I., Yatshyshyn S. P., Lutsyk Ya. T. (2017). Measuring Smart Means for Cyber-Physical Systems. Measuring Technology and Metrology: Issue 77, pp. 3–17.

2. What is Middleware?. Middleware.org. Defining Technology. (2008). Retrieved 2013-08-11.

3. Ad-hoc Sensor Networks. http://www.brunel.ac.uk/cedps/electronic-computer-engineering/research-activities/wncc/ad-hoc-sensor-networks

4. Dorozhovets M. M. (2014). Chapter "Conditioning the Sensors Signals". In "Sensors". Publishing House "Beskyd Bit", Lviv, Ukraine, pp.124–152.

5. Yatshyshyn S. P., Mykytyn I. P., Kravets I. P. (2010). Fire Sensors. Principles of Optimization of the Work and Algorithms of Decision Making, Fire Safety: Issue 17, pp. 14–19.

6. Oleskiv O. M., Mykytyn I. P. (2014). Analytical Review of Procedures and Methods of Metrological Testing of Software. Measurement Technology and Metrology: Issue 75, pp. 2–7.