

12. Nykolaychuk Y. Architecture and system characteristic of distributed computer network with autonomus sensor equipment / Y. Nykolaychuk, N. Krutskevych, O. Zastavnyy // Proc. of the International Conf. "Modern problems of radio engineering, telecommunications and computer science" TCSET 2006. – Lviv-Slavsko (Ukraine). – P. 394 – 398.

13. Nykolaychuk O. Y. Perspective Architecture and Components of Computer Networks / O Y. Nykolaychuk, N. Krutskevych O. Zastavniy, T. Grinchyshyn // Proc. Of the Second IEEE International Workshop on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications, IDAACS, Lviv, Ukraine, 2003.

14. Николайчук Я. М. Теорія джерел інформації / Николайчук Я. М. / – Тернопіль: ТНЕУ, 2008. – 536 с.

Надійшла 3.3.2011 р.

УДК 004.94

В.І. МЕЖУЄВ

Бердянський державний педагогічний університет

АНАЛІЗ ІНФОРМАЦІЙНОЇ ТЕХНОЛОГІЇ ПРЕДМЕТНО-ОРІЄНТОВАНОГО МАТЕМАТИЧНОГО МОДЕЛЮВАННЯ

У статті розглядається інформаційна технологія (ІТ) предметно-орієнтованого математичного моделювання (DSMM). Виділені класи ІТ та відповідних інструментів, риси яких інтегрують у собі засоби DSMM. Проведений порівняльний аналіз властивостей виділених класів інструментів з функціональністю засобів DSMM. Методом аналізу ієрархій розв'язана оптимізаційна задача вибору найбільш ефективної інформаційної технології.

The information technology (IT) of domain specific mathematical modelling (DSMM) is considered in the paper. The classes of IT and related software which features are integrated in DSMM tools are allocated. The comparative analysis of properties of the allocated classes of software with functionality of DSMM tools is given. The task of choosing the most effective information technology by the method of hierarchies analyses is solved.

Ключові слова: предметно-орієнтоване математичне моделювання.

Вступ

Дана стаття присвячена аналізу інформаційної технології (ІТ) предметно-орієнтованого математичного моделювання (DSMM, від англ. Domain Specific Mathematical Modelling). Особливістю ІТ DSMM є те, що вона включає риси багатьох підходів до комп'ютерного моделювання предметних областей (ПрО). Метою даної статті є доведення того положення, що цілеспрямована інтеграція властивостей різних ІТ дозволяє суттєво підвищити ефективність процесу моделювання ПрО. Довести ефективність нової ІТ і відповідних інструментальних засобів можна лише шляхом порівняння з існуючими підходами. Тому перша частина статті присвячена аналізу властивостей наявних ІТ моделювання ПрО та відповідних їм інструментальних засобів. На підставі аналізу виділяються критерії порівняння інструментів, встановлюються пріоритети даних критеріїв, що дозволяє обрати найбільш ефективну альтернативу.

Основна частина

DSMM є розширенням інформаційної технології предметно-орієнтованого моделювання (DSM, від англ. Domain Specific Modelling). Ключовою особливістю інструментів DSM є можливість створення специфічної для ПрО мови моделювання (метамоделі). Така властивість призводить до істотного підвищення ефективності процесу моделювання ПрО, тому що модель створюється у термінах ПрО саме фахівцем ПрО.

Відмінність DSMM від існуючої технології DSM полягає у можливості визначення на рівні метамоделі загального способу розв'язання класу задач (зокрема, генерації програмного коду). Крім того, DSMM інтегрує риси багатьох ІТ в єдиному інструменті, що призводить до поняття комплексних інструментальних засобів предметно-орієнтованого математичного моделювання.

Серед існуючих інструментів, риси яких інтегрують у собі засоби DSMM, можна виділити наступні класи:

1. Інструменти предметно-орієнтованого моделювання (MetaEdit+, Eclipse Modeling Framework, MS DSM Tools й ін.).
2. Інструменти моделювання ПрО (CASE-системи), що використовують універсальні мови (UML, SDL, SysML й ін.).
3. Системи автоматизації проектування (САПР, до яких відносяться класи CAD, CAE, CAM).
4. Системи автоматизації обчислень (MathCad, MatLab, Maple й ін.).
5. Інструменти автоматизації наукових досліджень – фізичного, інженерного та ін. моделювання й імітації (LabVIEW, MapleSim, MathModelica й ін.).
6. Системи керування змістом (Content Management Systems – CMS) – Plone, Drupal та ін.
7. Інструменти онтологічного моделювання (зокрема, Protege).
8. Системи автоматизації умовиводів (машини виведення, наприклад Fact++ плагін для Protege).

9. Валідатори моделей (TLC, SPIN, UPPAAL й ін.).
10. Синтаксичні аналізатори та генератори компіляторів (lex / yacc, flex / bison й ін.).
11. Віртуальні машини (CLR, JVM, Forth й ін.).
12. Інтегровані середовища розробки (IDE), особливо системи швидкісної розробки додатків (RAD).
13. Системи керування проектами (MS Project, OpenProj, TaskJuggler та ін.).
14. Інструменти моделювання бізнес процесів (BPWin, Rational Rose, Oracle Process Modeller й ін.).

Наведемо стисло характеристику властивостей даних класів інструментів з метою виділення функціональності, реалізованої у засобах DSMM.

Зазначимо, що аналізу інструментів предметно-орієнтованого моделювання (DSM) була присвячена стаття [1]. CASE-системи, що використовують UML та інші універсальні мови, були розглянуті нами у [2].

Системи автоматизації проектування

Інструменти DSMM мають множину схожих рис із системами автоматизації проектування. Помітимо, що основна мета створення та використання САПР – це підвищення ефективності етапів проектування виробів й підготовки виробництва. Підвищення ефективності досягається шляхом скорочення термінів, зниження собівартості робіт, підвищення якості й техніко-економічного рівня проектних робіт, скорочення витрат на натурне моделювання, виготовлення й випробування зразків та ін.

Для досягнення цієї мети як інструменти DSMM, так і САПР здійснюють підтримку користувача в розробці й генерації проектної документації, дозволяють уніфікувати й повторно використовувати проектні рішення, замінюють натурні розробку та випробування модельним експериментом. DSMM і САПР також містять як важливу складову програмне забезпечення, що реалізує математичні методи та алгоритми, необхідні для розв'язання задач автоматизації проектування. Помітимо, що як і DSMM, САПР включає лінгвістичне забезпечення – сукупність мов, що використовуються для подання інформації про проєктовані об'єкти, процеси, а також засоби проектування. Принципова відмінність полягає у тому, що DSMM надає можливість розробки власної (специфічної для ПрО) мови моделювання або проектування.

Системи автоматизації обчислень

Помітимо, що такі системи також часто відносять до класу САПР (так, у назві MathCAD навіть зафіксована аббревіатура Computer Aided Design, що є англійським еквівалентом САПР). Математичні пакети – це інтерактивні системи, що здійснюють підтримку розв'язання технічних і наукових задач шляхом застосування множини вбудованих методів й алгоритмів. Так, MathCAD дозволяє виконувати чисельні й символні обчислення, здійснювати операції зі скалярними величинами, векторами та матрицями. Таким чином, схожість із інструментами DSMM полягає в підтримці розв'язання задач шляхом застосування вбудованих у систему методів або ж навіть розробки власних алгоритмів, а основна відмінність – у використанні для моделювання ПрО універсальної математичної, а не предметно-орієнтованої мови. Зазначимо також, що для систем автоматизації обчислень є традиційним використання текстових, а не візуальних нотацій.

Інструменти інженерного моделювання й імітації

На відміну від попереднього класу програмних засобів, інженерні інструменти частіше використовують для моделювання ПрО деяку графічну (візуальну) мову. Розглянемо наприклад інструмент LabVIEW, в основі якого лежить основана на архітектурі потоків даних мова «G».

Модель в LabVIEW є віртуальним приладом і складається з блокової діаграми, що описує логіку роботи приладу і лицьової панелі, що описує його зовнішні інтерфейси. Такий підхід є близьким до сутності предметно-орієнтованого моделювання, за винятком того факту, що LabVIEW призначений для створення моделей фіксованої ПрО (а саме, електротехніки). Так, лицьова панель LabVIEW містить характерні для електротехніки засоби введення-виведення: кнопки, перемикачі, світлодіоди, шкали, інформаційні табло й т.п.

Ще однією схожістю з DSMM є те, що користувач LabVIEW за допомогою візуальної нотації (блокової діаграми) може визначити не лише структуру й функції ПрО, але й задавати спосіб розв'язання задач. До функціональних вузлів блокової діаграми відносяться як джерела й приймачі даних, так і управлючі структури, що є аналогами елементів мов програмування (умовний оператор, оператор циклу та ін.).

Також проведемо аналогію із системами інженерного моделювання, що використовують мову Modelica (OpenModelica, JModelica, MapleSim, MathModelica). Modelica є об'єктно-орієнтованою (ОО) і декларативною мовою моделювання складних гетерогенних (тобто таких, що складаються з різномірних частин – механічних, електричних, термічних й ін.) систем. Хоча Modelica й є ОО мовою, її не можна віднести до мов програмування. Класи Modelica не компілюються у код, а виконуються на спеціальній машині. У цьому полягає схожість з DSMM, де модель виконується на особливій (віртуальній) машині, що виконує конструкції у відповідності з правилами, зафіксованими на рівні метамоделі.

Іншою близькою до DSMM особливістю OpenModelica є можливість так званого кіберфізичного моделювання (що розроблюється в рамках ІТЕА2 OPENPROD проекту [3]). Мова йде про спеціальний профіль UML/Modelica, що поєднує похід системної інженерії (тобто розробку вимог, специфікацій та ін.) з фізичним моделюванням.

Системи керування змістом

Одним із принципів DSMM є можливість розробки моделей різних типів (як концептуальних, онтологічних так і формальних, математичних). Метамодель у DSMM задає структуру елементів предметної області. Тип цих елементів залежить від специфіки ПрО, задач, що розглядаються, наявних методів

розв'язання та інших чинників.

Для неформалізованих (або слабоформалізованих) предметних областей побудова концептуальних моделей є практично єдиною можливістю вивчення ПрО та розв'язання виникаючих задач. Так, виділення концептуальної схеми ПрО дозволяє побудувати структуру бази даних фактично без розуміння законів існування ПрО. Концептуальне моделювання у системній інженерії також розглядається як перший етап у розробці будь-якої системи (тут воно має сутністю формулювання вимог і специфікацій).

Про зв'язок DSMM з інструментами онтологічного моделювання (і можливість їхнього застосування в контексті системної інженерії) мова піде у наступному розділі статті. Проведемо паралель між DSMM і системами управління змістом (Content Management Systems, CMS), де типи змісту (контенту) використовуються для породження конкретних екземплярів змісту. Дані типи утворюють концептуальну схему, яку можна вважати метамоделлю для створення моделей ПрО.

Деякі з CMS (наприклад, Plone [4]) дозволяють редагувати концептуальну схему й навіть створювати нові типи контенту на основі базових понять системи. Таким чином, реалізується предметно-орієнтований підхід у контексті концептуального моделювання.

Відмінність DSMM від CMS полягає в можливості створення формальних моделей, а також застосування до даних моделей математичних методів розв'язання задач. Інший аспект полягає в тому, що метамодель в DSMM включає не тільки набір абстрактних понять (що слугують типами для породження екземплярів), але також визначає способи поєднання екземплярів понять (тобто граматику моделі). Наприклад, екземпляри визначених на рівні метамоделі понять "система" та "підсистема", можуть бути зв'язані на рівні моделі тільки відношенням "частина– ціле". Таким чином, структурування моделі здійснюється шляхом накладення обмежень на можливі відношення екземплярів типів метамоделі.

Інструменти онтологічного моделювання

У філософії онтологія визначається як учіння про буття. Інше (і більш вузьке) визначення онтології дається в Semantic Web, де воно означає формальну (звичайно, основану на логіці першого порядку) специфікацію концептуалізації [5].

Таким чином, онтологічне моделювання основане на концептуалізації ПрО, тобто визначенні найважливіших понять ПрО й формалізації відношень між ними. Для цього використовують мови онтологій, наприклад OWL (Web Ontology Language). У статті [6] нами було доведено, що побудова онтологій є окремим випадком IT DSMM. У цьому контексті метамодель розглядається як онтологія верхнього рівня й використовується для створення предметно-орієнтованих онтологій ПрО.

Однак на відміну від DSMM, онтологічна модель у Semantic Web є повністю дескриптивною. Для формалізації відношень між поняттями ПрО застосовують логіку першого порядку, а частіше її DL (Description Logic) підмножину. Даний підхід дозволяє довести деякі властивості онтологічної моделі ПрО, однак не дає можливості визначити спосіб розв'язання задачі.

Метамодель в DSMM задає не тільки структуру типів (концептів) ПрО, але й дозволяє визначити спосіб розв'язання специфічної для предметної області задачі. Більше того, метамодель у DSMM може бути використана для структурування елементів ПрО довільного виду, у тому числі таких, що мають строге математичне визначення (наприклад, фізичних величин). IT DSMM також дозволяє здійснити різні способи структурування властивостей, що відбивають знання ПрО. Так, у статті [7] нами був розглянутий підхід, де висловлювання про ПрО (силогізми) розподілені на векторах у логічному просторі.

Великий інтерес також має можливість використання інструментів онтологічного моделювання (та відповідно IT DSMM) у контексті системної інженерії. Взагалі кажучи, основну ідею системної інженерії можна сформулювати так: перш ніж бути реалізованою, система повинна бути специфікована. Під специфікацією розуміють твердження про необхідні властивості, структуру й поведінку майбутньої системи.

Онтології дозволяють формалізувати специфікації систем, що традиційно здійснюється природною мовою. Необхідність цього впливає з відомих вад природної мови, а саме її надмірності та неоднозначності (залежності від контексту). Крім того, лише до формальних специфікацій застосовне поняття адекватності, що дозволяє перевірити, відповідає або не відповідає побудована модель системи вираженням у вигляді онтологій специфікаціям.

Системи автоматизації умовиводів

Таким чином, онтологічний підхід до моделювання ПрО є важливим окремим випадком DSMM. Цінність формального подання концептуалізації ПрО полягає в можливості використання спеціальних програм, які відносяться до класу систем автоматизації умовиводів (CAU). Наприклад, у випадку Protege використовується інструмент FACT++, що дозволяє здійснити перевірку різних властивостей OWL-DL онтологій (найчастіше це коректність ієрархії концептів онтології, зв'язаних відношенням Is-A). Природно, що існуючий CAU можуть бути також використані в контексті IT DSMM.

Валідатори моделей (TLC, SPIN, UPPAAL й ін.)

Як було зазначено нами вище, формальна специфікація системи надає можливість її валідації. Для цього використовують спеціальні інструменти – валідатори (англ. *checkers*), що перевіряють всі можливі стани системи на відповідність її специфікаціям.

Найважливішою властивістю інструментів DSMM є можливість формальної перевірки побудованих з їх допомогою моделей ПрО.

Така властивість є наслідком двох причин:

- способу побудови алфавітів мов DSMM;
- формальної специфікації граматики мов.

Символи алфавітів мов DSMM на рівні моделі є екземплярами типів метамоделі та мають власний набір значень атрибутів, що відбиває поточний стан ПрО. Граматику мови DSMM можна розглянути як певний інваріант – тобто властивість, якій повинні відповідати всі можливі стани системи. Таким чином, користувач інструменту DSMM будує модель ПрО у відповідності до граматики мови, тип самим гарантуючи валідність моделі.

Генератори компіляторів

Ключовою особливістю інструментів DSMM є можливість розробки мови моделювання (метамоделі) у рамках понять більш високого рівня абстракції (мета-метамоделі). Помітимо, що даний підхід не є принципово новим. Існують формальні нотації (найбільш відомою з яких є форма Бэкуса-Наура, BNF), що дозволяють виразити граматику мови програмування (або моделювання). Така граMATика є входом для спеціального інструменту, що називається генератором компіляторів (наприклад, lex / yacc, flex / bison та ін.), результатом роботи якого є синтаксичний аналізатор і компілятор даної мови.

Строго говорячи, синтаксичний аналізатор є частиною генератору компіляторів. Синтаксичний аналіз мови – це процес зіставлення лінійної послідовності елементів мови з її формальною граMATикою. Результатом роботи такого інструменту є структура даних (звичайно, деревоподібна), що відбиває синтаксис вхідної мовної конструкції.

Семантика мови виражається шляхом зіставлення з кожною синтаксичною конструкцією процедури генерації коду. Така процедура викликається щораз при розпізнанні синтаксичним аналізатором даної конструкції мови у вихідному тексті.

Аналогічний механізм використовується в інформаційній технології DSMM. Однак, генерація коду є лише окремим випадком функціональності інструментів DSMM, де у відповідність синтаксичним конструкціям мови може бути поставлена будь-яка програмна функція. Таким чином, IT DSMM дозволяє на рівні метамоделі (граматики мови моделювання) визначити спосіб розв'язання класу задач ПрО (модель у цьому випадку є повністю декларативною).

Іншим аспектом є виразність існуючих формальних нотацій, призначених для опису граMATик мов програмування та моделювання. Помітимо, що у випадку використання візуальних нотацій, мовна конструкція не є лінійною послідовністю символів, а може мати два або більшу кількість вимірів. Дослідженню виразності візуальних мов та відповідних граMATик будуть присвячені наші наступні роботи.

Віртуальні машини

Генератори компіляторів перетворюють вихідний текст програми в код, що буде виконуватися на деякій обчислювальній машині. В інструментах DSMM модель ПрО виконується усередині самого інструмента DSMM. Така властивість приводить до визначення інструмента DSMM як *віртуальної машини*, що емулює поведінку деякого реального обчислювального пристрою.

Модель ПрО виконується в контексті мета-метамоделі за правилами, зафіксованим на рівні метамоделі. Наприклад, модель визначає структуру програмного додатка. У цьому випадку метамодель задає граматику моделі (можливі способи з'єднання її елементів), а також методи опрацювання моделі, зокрема, правила обходу моделі, що приводять до генерації коду програмного додатка.

Таким чином, метамодель задає структуру обчислювального процесу, що виконується на віртуальній машині DSMM для структур даних, визначених на рівні моделі. Даний процес також використовує ресурси (бібліотеки програмних функцій), визначені на рівні мета-метамоделі.

Інтегровані середовища та системи швидкісної розробки додатків

Під інтегрованим середовищем розробки (Integrated Development Environment, IDE) розуміється система інструментальних засобів, призначених для розробки програмного забезпечення. Властивість «інтегрованості» визначає, що IDE включає множину інструментів, до яких традиційно відносяться текстовий редактор, компілятор та налагоджувач програми. Сучасні IDE також мають засоби для візуальної розробки графічних інтерфейсів, використання яких підвищує швидкість створення додатків у десятки разів. У цьому випадку IDE переходить у розряд систем швидкісної розробки додатків (Rapid Application Development, RAD). Засоби DSMM також включають множину інструментів, що слугують для підтримки користувача в процесі розробки моделі ПрО. Із системами RAD засоби DSMM поєднує можливість швидкісної візуальної розробки моделі ПрО.

Системи керування проектами

Такі системи використовуються для підтримки діяльності менеджерів при аналізі обсягів робіт, плануванні етапів проекту, відстеженні прогресу в здійсненні проекту, розподілі ресурсів і т.д. Завдяки можливості розробки довільної мови, засоби DSMM здійснюють підтримку користувача й у такий ПрО, як керування проектами. Найважливішою відмінністю інструментів DSMM є можливість не лише моделювання процесів, але й організації діяльності користувачів відповідно до визначеної моделі процесу.

Аналіз ефективності інформаційних технологій моделювання ПрО

Таким чином, DSMM містить у собі риси множини різних IT комп'ютерного моделювання ПрО. Специфіка даних інформаційних технологій істотно залежить від ПрО, що розглядаються, а також класів задач, які необхідно розв'язати. У той же час можна відібрати близькі за сутністю інформаційні технології і

виділити в них інваріанти властивостей та функціональності (наприклад, застосування універсальних або розробка предметно-орієнтованих мов, використання візуальних або текстових нотацій, здійснення концептуального або формального моделювання й ін.).

Виділення й аналіз даних критеріїв дозволяє вирішити оптимізаційну задачу прийняття рішення, метою якого є вибір найбільш ефективної інформаційної технології комп'ютерного моделювання ПрО. У даному розділі статті наведено розв'язання даної задачі методом аналізу ієрархій (МАІ) [8; 9].

МАІ надає процедуру для декомпозиції визначаючих суть проблеми критеріїв, з наступною їх обробкою шляхом попарних порівнянь, що дозволяє чисельно виразити пріоритетність критеріїв та, власне, знайти рішення. Таким чином, розв'язання проблеми методом МАІ є процесом поетапного встановлення пріоритетів критеріїв. При цьому критерії повинні охоплювати всі важливі характеристики проблеми, необхідні для її розв'язання.

Першим етапом МАІ є декомпозиція проблеми у вигляді трьохрівневої ієрархії, починаючи з мети (вершина ієрархії), через критерії оцінки до найнижчого рівня (переліку альтернатив).

Постановка мети застосування МАІ. Із множини схожих за сутністю інформаційних технологій комп'ютерного моделювання ПрО вибрати ІТ, що забезпечує найкращі оцінки відповідно до МАІ.

Відібраними **альтернативами вибору** є наступні ІТ (та відповідні їм інструментальні засоби):

А 1. ІТ моделювання ПрО, що використовують універсальні мови (CASE-UML).

А 2. ІТ предметно-орієнтованого моделювання (DSM).

А 3. ІТ предметно-орієнтованого математичного моделювання (DSMM).

У результаті експертної оцінки була обрана множина критеріїв оцінки властивостей інформаційних технологій комп'ютерного моделювання ПрО. Дані критерії були розбиті на кілька груп, що відбивають різні аспекти аналізованих ІТ (лінгвістичні, процесуальні, функціональні). Це було зроблено для відповідності сформульованому в [8; 9] критерію погодженості суджень.

Помітимо також, що відповідно до МАІ, кількість порівнюваних об'єктів повинна бути достатньо малою, звичайні межі – це 7 ± 2 одиниць. Тому в списку критеріїв були залишені лише найважливіші й відкинуті другорядні.

У даній роботі ми розглянемо критерії, які визначають часові параметри ефективності застосування ІТ:

К 1. Швидкість розробки моделі.

К 2. Швидкість розробки метамоделі (або ж предметної адаптації універсальної мови моделювання).

К 3. Швидкість зміни моделі у відповідь на запити.

К 4. Швидкість переходу між етапами діяльності (від вимог до архітектури, від архітектури до реалізації та ін.).

К 5. Тривалість етапу навчання мові моделювання.

На другому етапі МАІ особою, що приймає рішення (ОПР) були побудовані матриці для порівняння відносної важливості критеріїв з метою встановлення їх пріоритетів. Методика обчислення вектору пріоритетів для матриці попарних порівнянь надана у [8; 9].

Отримані значення компонентів вектору локальних пріоритетів надали можливість ранжувати критерії відповідно до оцінок ОПР за убуванням отриманих ваг:

1. Швидкість розробки моделі (вага – 0,3767285).

2. Швидкість зміни моделі у відповідь на запити (вага – 0,268545).

3. Швидкість переходу між етапами діяльності (вага – 0,1515467).

4. Швидкість розробки метамоделі (вага – 0,1227431).

5. Тривалість етапу навчання мові моделювання (вага – 0,0804366).

Побудова матриць попарних порівнянь третього рівня

Для кожного критерію були проведені попарні порівняння альтернатив і був здійснений синтез локальних пріоритетів (ми упустимо розрахунки з метою скорочення об'єму статті).

Синтез глобальних пріоритетів для ІТ моделювання ПрО

На основі матриць попарних порівнянь третього рівня були розраховані значення компонент вектора глобальних пріоритетів та здійснений синтез глобальних пріоритетів для різних ІТ моделювання ПрО.

Результати обчислень можна трактувати як значення функції корисності для кожної з альтернатив. На першому місці виявилася ІТ предметно-орієнтованого математичного моделювання, що випередила найближчого сусіда «ІТ предметно-орієнтованого моделювання» на $(0,4726-0,3410) \times 100 \% = 13,16 \%$.

Висновки

1. Визначені найважливіші властивості інструментів предметно-орієнтованого математичного моделювання. Засоби DSMM включають риси множини ІТ комп'ютерного моделювання предметних областей, тобто є комплексними інструментальними засобами.

2. Виділені класи існуючих інструментів, риси яких інтегрують у собі засоби DSMM. Проведений порівняльний аналіз властивостей виділених класів інструментів з функціональністю засобів DSMM.

3. Виділені близькі за сутністю до DSMM інформаційні технології комп'ютерного моделювання ПрО та виділені критерії їх порівняння. Методом аналізу ієрархій розв'язана оптимізаційна задача прийняття рішення, метою якого є вибір найбільш ефективної інформаційної технології. Найкращі оцінки

Література

1. Межуєв В. І. Технології та інструментальні засоби предметно-орієнтованого моделювання / В. І. Межуєв // Системи управління, навігації та зв'язку. – 2010. – № 3 (15). – С. 102– 106.
2. Межуєв В. І. Доцільність застосування UML для моделювання систем озброєння та військової техніки / В. І. Межуєв // Системи озброєння та військова техніка. – 2010. – № 2 (22). – С. 122– 126.
3. Openprod project [Електроний ресурс]. – Режим доступу: www.openprod.org
4. Plone CMS [Електроний ресурс]. – Режим доступу: www.plone.org
5. Gruber T. R. A translation approach to portable ontologies. Knowledge Acquisition, 5 (2). 1993. – pp. 199– 220.
6. Межуєв В. І. Онтологические модели систем и процесса системной инженерии / В. І. Межуєв // "Искусственный интеллект". – 2010. – № 4. – С. 606– 616.
7. Mezhuuev V. Vector logic: theoretical principles and practical implementations // Вісник ЗНУ: збірник наукових статей. Фізико-математичні та технічні науки. – Запоріжжя: ЗНУ, 2006. – С. 91– 97.
8. Саати Т. Аналитическое планирование. Организация систем / Т. Саати, К. Кернс; [пер. с англ.] – М.: Радио и связь, 1991.
9. Шикин Е. В. Математические методы и модели в управлении: [учеб. пособие] / Е. В. Шикин, А. Г. Чхартишвили. – М.: Дело, 2000. – 440 с.

Надійшла 7.3.2011 р.

УДК 621.396.7

Ю.М. БОЙКО, В.М. СВОРІНЬ
Хмельницький національний університетОСОБЛИВОСТІ РЕАЛІЗАЦІЇ АЛГОРИТМІВ АДАПТИВНОЇ
ФІЛЬТРАЦІЇ В СИСТЕМАХ ОБРОБКИ СИГНАЛІВ

Проведено огляд алгоритмів адаптивної фільтрації в системах обробки сигналів. Розглянуто особливості використання LMS та RLS алгоритмів адаптивної фільтрації, алгоритму фільтрації Калмана в системах обробки сигналів. Розглянуто математичні моделі перелічених алгоритмів фільтрації та досліджено збіжність алгоритмів в середовищі Matlab Simulink. Досліджено особливості використання перелічених алгоритмів в системах зв'язку при побудові еквалайзерів, з допомогою середовища Matlab Simulink.

The review of algorithms of adaptive filtration is conducted in the systems of treatment of signals; to consider the features of the use of LMS, RLS, to the algorithm of filtration of Kalman in the systems of treatment of signals. The mathematical models of the transferred algorithms of filtration and probed meets of algorithms are considered in the environment of Matlab Simulink. The features of the use of the transferred algorithms are probed in communication networks at the construction of equalize, with the help of environment of Matlab Simulink.

Ключові слова: адаптивна фільтрація, метод найменших квадратів, адаптивна система.

Вступ

Досягнутий останнім часом прогрес в розробці і виробництві мікросхем призвів до створення дуже компактних, економічних і надійних пристроїв обробки сигналів, конкуруючих з біологічними нейронними системами за розмірами і таких, що перевершують біологічні системи по швидкодії механічного впливу. В результаті цього значно розширилася сфера їх застосування у всіх видах цифрової обробки сигналів, у тому числі адаптивної обробки [1]. В даний час адаптивні системи застосовуються в таких областях, як зв'язок, радіолокація, гідролокація, сейсмологія, проектування механічних систем, навігація і біомедична електроніка. Основною властивістю адаптивної системи є змінне в часі функціонування з саморегулюванням. Необхідність такого функціонування очевидна з наступних міркувань. Якщо розробник проектує «незмінну» систему, яку він вважає оптимальною, то це означає, що розробник передбачає всі можливі умови на її вході, по меншій мірі у статистичному сенсі, і розраховує, що система буде працювати при кожній з цих умов. Далі розробник обирає критерій, за яким має оцінюватися функціонування, наприклад середня кількість помилок між вихідним сигналом реальної системи і вихідним сигналом деякої обраної моделі або «ідеальної» системи. Нарешті, розробник обирає систему, яка виявляється кращою відповідно до встановленого критерію функціонування, зазвичай, з деякого апріорно обмеженого класу (наприклад, з класу лінійних систем). Однак, у багатьох випадках весь діапазон вхідних умов може бути не відомий точно навіть у статистичному сенсі або умови можуть час від часу змінюватися. Тоді адаптивна система, використавши регулярний процес пошуку, постійно шукає максимум в межах допустимого класу можливостей і таким чином встановлює переваги в порівнянні з незмінною системою. Адаптивні системи за своєю природою мають бути змінними в часі і нелінійними. Наприклад, якщо сигнал подається на вхід адаптивної системи для визначення властивостей по її відгуку, то система адаптується до цього певного вхідного сигналу і тим самим змінює власну структуру. Таким чином, адаптивні системи по суті важко