

УДК 001:681.3

О. Д. Азаров, О. Н. Романюк

ПРО ОРГАНІЗАЦІЮ НАУКОВОЇ РОБОТИ В ІНСТИТУТІ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ ТА КОМП'ЮТЕРНОЇ ІНЖЕНЕРІЇ

Вінницький національний технічний університет, Вінниця

Ця стаття започатковує перший номер журналу. Її присвячено аналізу наукової діяльності інституту інформаційних технологій та комп'ютерної інженерії (ІнІТКІ) протягом останніх років. Автори звертають увагу на дещо нетрадиційний підхід щодо структури і компонування цієї праці і разом з тим вважають доцільним саме такий підхід, оскільки матеріал, який наводиться, сприяє дискусії щодо організації наукової роботи у вищих навчальних закладах.

Інститут інформаційних технологій та комп'ютерної інженерії було створено у 2002 році на базі факультету інформаційних технологій та комп'ютерної інженерії (ФІТКІ). До його складу увійшли факультет комп'ютерного інтелекту (ФКІ) та факультет комп'ютерних систем та мереж (ФКСМ).

Факультет комп'ютерного інтелекту складається з трьох кафедр: двох випускних кафедр – програмного забезпечення (ПЗ), інтелектуальних систем (ІС) та кафедри фундаментальної підготовки – вищої математики (ВМ).

До складу факультету комп'ютерних систем та мереж входять дві випускні кафедри – обчислювальної техніки (ОТ) та захисту інформації (ЗІ), кафедра іноземних мов (ІМ), а також факультет підвищення кваліфікації з сучасних комп'ютерних технологій та науково-навчальний комп'ютерний центр перспективних інформаційних технологій (ННКЦ ПІТ).

В ІнІТКІ склалася ефективна система науково-дослідної діяльності викладачів, співробітників та студентів, сформувалися актуальні наукові напрямки досліджень, є в наявності значний потенціал науковців, до складу якого входять 6 докторів наук, 36 кандидатів, а також 32 та 11 пошукачів на здобуття наукових ступенів кандидата та доктора відповідно.

На випускних кафедрах, які входять до складу інституту, діють і формуються наукові школи з актуальних проблем інформаційних технологій та обчислювальної техніки. Так, під керівництвом професора Азарова О. Д. на кафедрі ОТ виконуються дослідження з проектування та розробки високопродуктивних засобів перетворення аналогових сигналів на основі інформаційної надлишковості. Крім цього спільно з доцентом Яремчуком Ю. С. здійснюються дослідження та розробка засобів криптографічного захисту інформації. На кафедрі ПЗ під керівництвом професора Петуха А. М. проводиться розробка нових методів подання та відображення інформації, а також засобів підвищення інтерактивності інформаційної взаємодії. На кафедрі ІС під керівництвом професора Юхимчука С. В. здійснюється розробка методів аналізу та синтезу робастних лінійних та нелінійних систем. Під керівництвом професора Клочка В. І. на кафедрі ВМ проводиться розробка теоретичних основ комп'ютерно-орієнтованих дидактичних систем організації та активізації навчальної діяльності. На кафедрі ЗІ під керівництвом професора Лужецького В.А. виконуються дослідження щодо криптографічного захисту інформації. Отже, на кафедрах інституту сформовано систему науково-дослідної роботи викладачів, наукових співробітників, аспірантів та студентів, яка є складовою частиною процесу формування майбутніх фахівців і базою для підготовки кадрів вищої наукової кваліфікації.

Цей напрямок діяльності постійно знаходиться під контролем керівництва інституту і дає свої позитивні результати. За три останні роки захищено дві докторські дисертації (Роїк О. М., Лужецький В. А.) та 8 кандидатських дисертацій: Біліченко Н. О., Яремчук Ю. С., Шапошніков О. В., Суприган О.І., Сачанюк-Кавецька Н. В., Архипчук О. А., Чорний А. В., Хазін М.Б. Активно працюють над докторськими дисертаціями доценти Романюк О. Н., Месюра В. І., Перевозніков С. І., Дереч В. Д., Яремчук Ю. С., Хошаба О. М., Майданюк В. П., Захарченко С. М.

На всіх випускних кафедрах інституту є аспірантура. В аспірантурі станом на 2004 рік навчається 15 осіб. Керують аспірантурою в ІнІТКІ провідні вчені. Це завідувачі кафедр Азаров О. Д., Клочко В. І., Петух А. М., Юхимчук С. В. Лужецький В. А. та викладачі: доценти Колодний В. В., Месюра В. І., Романюк О. Н., Обідник Д. Т., Семеренко В. П., Кондратенко Н. Р., Яремчук Ю. С., Хошаба О. М.

Провідні науковці інституту беруть участь у роботі спеціалізованих рад по захисту кандидатських та докторських дисертацій. Так, професор Азаров О. Д. є головою спеціалізованої ради Д 05.052.02 та членом спеціалізованої ради Д 05.052.01. Професор Петух А. М. є членом спеціалізованих рад Д 05.052.01, Д 05.052.02, професор Юхимчук С. В. є членом спеціалізованих рад Д 05.052.01,

Д 05.052.02, професор Ключко В.І є членом спеціалізованої ради Д 26.053.03, доцент Захарченко С. М. є вченим секретарем спеціалізованої ради Д 05.052.02.

Викладачі та співробітники ІніТКІ виконують значну науково-дослідну роботу, обсяг фінансування якої в 2001-2003 роках становив 169,7 тис. грн.. За результатами наукових досліджень за три останні роки опубліковано 458 наукових праць, включаючи 11 монографій, 47 навчальних посібників. Отримано 16 патентів України.

Наукова робота на факультетах ІніТКІ проводиться в режимі активного співробітництва з провідними навчальними закладами та науковими центрами України, зокрема, в містах: Києві, Харкові, Львові, Одесі, Донецьку, Чернівцях, Чернігові, Хмельницькому та інших. Підтримуються тісні зв'язки з науковцями та підприємцями в галузі перетворення форми інформації та цифрового оброблення інформації, комп'ютерної графіки, розпізнавання зображень, захисту інформації, контролю та діагностики обчислювальних систем та інших. Налагоджено співробітництво із Всесвітнім інститутом інженерів з електроніки та електротехніки IEEE. Підтримуються також наукові зв'язки з університетами, науковими установами та фірмами багатьох країн СНД, а також США, Канади, Німеччини, Польщі, Швеції, Румунії. За останній рік укладено договори про науково-технічну співдружність з академією телекомунікацій інформатики Росії, Донецьким технічним університетом, Миколаївським національним університетом кораблебудування імені адмірала Макарова, Бакинським державним університетом.

Науковці ІніТКІ за три останні роки зробили 152 доповіді на міжнародних і Всеукраїнських конференціях та симпозиумах. Активну участь беруть викладачі, аспіранти і студенти на щорічних науково-технічних конференціях професорсько-викладацького складу ВНТУ. За три роки в секціях ІніТКІ було заслухано 521 доповідь.

Під керівництвом професора Петуха А. М. на кафедрі ПЗ впроваджують у навчальний процес нові технології колективної взаємодії, які дозволяють суттєво підвищити активність студентів під час навчання при використанні обмеженого набору технічних засобів. Уже функціонують два таких класи.

Науковці ІніТКІ не тільки беруть активну участь у роботі національних та міжнародних виставок, конференцій і симпозиумів, а також регулярно організують проведення подібних заходів. ІніТКІ кожних два роки проводить міжнародну конференцію "Інтернет–Освіта–Наука". Так, з 5 по 9 жовтня 2004 року відбулася четверта міжнародна конференція "Інтернет–Освіта–Наука–2004". На ній було подано 229 доповідей від 362 представників 20 країн світу. Під час конференції було проведено практичні семінари із систем дистанційного навчання та Web-програмування.

На випускних кафедрах ІніТКІ регулярно проводяться засідання наукових семінарів. Так, на кафедрі ОТ організовано міжрегіональний семінар Наукової Ради НАН України "Технічні засоби захисту інформації".

В інституті на високому рівні поставлено науково-дослідну роботу студентів. За три останні роки переможцями Всеукраїнського конкурсу студентських наукових робіт стали Хазін М.Б., Сокирук В.В., Моторнюк Р. Л., Чорний А. В., Гільманов Р. Р., Степанюк М. А., Табачник Ю. Я., Зігангірова Ю.М., Коржик П. В., Янюк В. А., Лисак Н. В., Войцех О. А. За три останні роки студентами ІніТКІ опубліковано 126 наукових праць, отримано 10 патентів. За звітний період на щорічних НТК професорсько-викладацького складу ВНТУ було заслухано 334 доповіді студентів. За три останні роки магістратуру закінчили 41 особа. У 2003 році вперше в історії інституту було здійснено випуск 3 магістрів, які навчалися за напрямком - менеджерська діяльність у галузях народного господарства зі спеціальностей 8.091501, 8.080403. Кафедрами інституту щорічно проводяться наукові конференції та предметні олімпіади з програмування, англійської мови, електронних обчислювальних машин, інформатики, системного програмування та вищої математики. За три останні роки в предметних олімпіадах ІніТКІ взяло участь 494 студенти.

У 2001 році студенти Євген Цвігун і Андрій Катасонов посіли відповідно перше та третє місця в міжнародній олімпіаді з програмування асоціації МАРК. У 2001 році на Всеукраїнській олімпіаді з програмування в командному турі друге місце виборола команда студентів з ІніТКІ у складі Олексія Харькова (2КІ-98), Євгена Цвігуна (1КІ-96) та Миколи Степанюка (1ПТІ-97), третє місце посіла команда у складі Павла Коржика (1ПТІ-99), Андрія Катасонова (2КІ-99) та Володимира Сорокіна (1ПТІ-99), в особистому турі змагання друге місце посіли Олексій Харьков (2КІ-98) та Павло Коржик (1ПТІ-99). На Всеукраїнській олімпіаді з математики у 2001 р. серед студентів вищих технічних закладів студент ІніТКІ Олексій Харьков (2КІ-98) посів третє місце. У 2002 році студентів ІніТКІ визнано переможцями відкритого Всеукраїнського чемпіонату з комп'ютерних ігрових і навчальних програм "Екософт-2002", а студент Горячкін Д. (гр. 1ПТІ-98) посів третє місце у Всеукраїнській олімпіаді з технічної діагностики обчислювальних систем та мереж.

Завдяки високим результатам студентів ВНТУ в олімпіадах із програмування, нашому університету з 2001 р. було доручено проведення Всеукраїнської олімпіади з програмування. У 2001 р. команди ІніТКІ посіли перше і третє місця у фіналі олімпіади, а у 2002 році – друге й шосте. Крім того, у півфіналі Всеукраїнської олімпіади 2002 р., який відбувся у Хмельницькому, три наші команди посіли

місяця з другого по четверте, пропустивши вперед лише господарів олімпіади, а в особистому турі серед перших семи місць всі, крім третього, посіли представники ВНТУ.

Починаючи з 1996 року, студенти ІНТКІ беруть активну участь у півфінальних змаганнях першості світу з програмування по південно-східному Європейському регіону, які вже 27 років проводяться Асоціацією комп'ютерної техніки (АСМ). У 2002 році команда ІНТКІ у складі Богдана Власюка, Олексія Харькова та Павла Коржика посіла шосте місце серед 48 кращих команд з 13 країн південно-східної Європи, призерів національних олімпіад. При цьому протягом двох з половиною годин команда лідирувала у змаганнях і лише за останні півгодини не досягла повної перемоги.

Протягом трьох останніх років студенти ІНТКІ виборювали призові місця на олімпіаді з сучасних програмних засобів проектування цифрових схем, яка проводилася фірмою Aldec(США). Переможці олімпіади Цвігун Євген (2-КС-96), Коцюба Сергій (1-КС-97), Харьков Олексій (1КС-98) проходили стажування у філіях фірми в Польщі (м.Катовіце).

У листопаді 2002 року студенти Олександр Шаригін і Владислав Янюк стали переможцями відкритої Всесибірської олімпіади з програмування інтелектуальних ігор. Програми, розроблені 20 командами-учасниками, змагалися між собою за круговою системою на восьми вхідних наборах даних. Програма наших студентів отримала перемогу в усіх 19 зустрічах. Команда студентів-програмістів нашого університету у складі Богдана Власюка (2ІС-00), Павла Коржика (1ІС-99) і Дмитра Коржика (ІС-02) на початку квітня 2003 року виборола перше місце у півфіналі першості України з програмування. На міжнародному студентському конкурсі "Hard&Soft-2003" команда ІНТКІ посіла четверте місце та отримала спеціальний приз за краще апаратне вирішення. Тренував команду доц. Месюра В. І. У 2003 році на п'ятій Всеукраїнській виставці-конкурсі "Комп'ютер і студент" студент Горячкін Д.Ю. посів третє місце в номінації " Навчальні програми й тренажери". У квітні 2004 року команда студентів ІНТКІ у складі Богдана Власюка (2 ІС-00), Дмитра Білоуса (ІС-02), Павла Зарезенка (ІС-03) посіла перше абсолютне місце у півфіналі Всеукраїнської студентської олімпіади з програмування, в якій узяли участь 124 команди з провідних університетів України. Команда Дмитра Коржика (ІС-02), Павла Коржика (ІС-м) і Юрія Коновалюка (ІС-02) посіла в цій олімпіаді почесне четверте місце. У Кубку України з програмування серед студентських команд, що відбувся 22-30 серпня 2004 році у м. Одесі команда ІНТКІ посіла друге місце. У вересні 2004 року у ВНТУ проходила Всеукраїнська олімпіада з програмування, яка мала статус чвертьфіналу світу. В олімпіаді взяли участь 32 команди . Команда ВНТУ у складі братів Коржиків та Дмитра Білоуса посіла друге місце, причому поступилася першому місцю тільки через незначну затримку в розв'язанні задач.

Робота з веб-дизайну студента Романюка С. О. на міжнародному конкурсі в США пройшла три відбіркові тури й увійшла до списку 200 кращих робіт студентів світу з понад 5000 робіт, які було представлено на конкурс. У 2004 році на шостій Всеукраїнській виставці-конкурсі "Комп'ютер і студент" студента Романюка С. О. визнано також переможцем у номінації "Найкраща графічна програма".

Слід відзначити, що підготовку до олімпіад в ІНТКІ проводять планово, протягом року, для чого залучаються кошти з ННКЦ ППТ.

Інститут щорічно проводить обласні конкурси серед студентів і учнів з веб-дизайну. За останні три роки на конкурс було подано понад 500 робіт. У конкурсах з веб-дизайну виборювалися призові місця в трьох номінаціях –"Найбільш актуальна робота", "Найкраще графічне оформлення", "Найкраща програмна реалізація". У 2004 році студентами ІНТКІ було подано на конкурс 94 роботи. Переможцями конкурсу у різних номінаціях визнано відразу вісім студентів ІНТКІ: Романюка Сергія, Павлюк Оксану, Кучера Константина, Мачушенка Дмитра, Черевиченка Олександра, Захаренкова Сергія, Фефермана Олега, Клецка Павла.

Студентам, які не тільки вчаться на "відмінно", а також активно займаються науковою роботою та беруть участь у громадському житті університету призначаються стипендії різних рівнів. Так, стипендія Президента України призначалась студентам Месюрі Н. В., Войцех О. А. Студентам Бежнару Д. В., Харьковову О.М. призначалася стипендія Ученої Ради ВНТУ. Студенту Улучу М. В. призначалася стипендія Кабінету міністрів України.

Керівництво ІНТКІ запрошує науковців ВНТУ, а також вищих наукових закладів і науково-дослідних організацій України, ближнього і дальнього зарубіжжя до активної участі в роботі журналу.

Азаров Олексій Дмитрович, директор інституту інформаційних технологій та комп'ютерної інженерії, завідувач кафедри обчислювальної техніки, д.т.н., професор, Вінницький національний технічний університет, вул. Хмельницьке шосе, 95, Вінниця, 21021, Україна.

Романюк Олександр Никифорович, заст. директора з наукової роботи інституту інформаційних технологій та комп'ютерної інженерії, к.т.н., Вінницький національний технічний університет, вул. Хмельницьке шосе, 95, Вінниця, 21021, Україна.

УДК 656.052:681.51.54

В. М. Дубовой, О. О. Ковалюк

ПРИЙНЯТТЯ РІШЕНЬ У РОЗПОДІЛЕНИХ СИСТЕМАХ ЗА УМОВ КОМБІНОВАНОЇ НЕВИЗНАЧЕНОСТІ

Вінницький національний технічний університет, Вінниця

Вступ

Актуальність. Останнім часом все більша увага приділяється розробці методів прийняття рішень в розподілених системах. Прикладами таких систем є системи телекомунікацій і зв'язку, керування транспортом, виробничі системи тощо. Розподілені системи переважно працюють в умовах невизначеності. Ця невизначеність пов'язана з неможливістю одночасного і точного визначення параметрів стану в усіх точках розподіленої системи і може мати стохастичну або нечітку природу. Стохастична невизначеність полягає у використанні випадкових величин, які є результатом вимірювань або накопичуються протягом певного часу. Нечітка невизначеність пов'язана з використанням функцій належностей, що формуються на основі суб'єктивних тверджень експертів. На сьогоднішній день розроблені методи прийняття рішень в умовах нечіткої або стохастичної невизначеності [2,4]. Проте існують системи з комплексною невизначеністю, в яких для прийняття рішень необхідно використовувати як нечіткі, так і стохастичні дані.

Таким чином, постає **проблема** прийняття рішень в умовах комбінованої стохастичної і нечіткої невизначеності, яка досліджена недостатньо і потребує подальшого вивчення.

Для розв'язання проблеми використовують переважно два підходи: представлення нечіткої невизначеності як стохастичної і використання теорії статистичних розв'язків [4,5] або представлення стохастичної невизначеності як нечіткої і використання нечіткої логіки [2,3]. Проте оптимальні розв'язки, отримані за допомогою цих підходів, можуть відрізнятися між собою.

Метою роботи є розробка методу прийняття рішень в системах керування розподіленими системами в умовах комбінованої невизначеності.

Для розв'язання поставленої задачі проведено дослідження результатів використання відомих методів на прикладі задачі регулювання перехрестям, яка описана в [7].

Задача керування перехрестям з одностороннім рухом полягає у зміні сигналів світлофора таким чином, щоб час затримки транспортних засобів (ТЗ) на перехресті був мінімальним. Нехай вхідними величинами задачі є :

- інтенсивність прибуття ТЗ на перехрестя по центру, задана у стохастичній формі;
- інтенсивність прибуття ТЗ справа, задана у нечіткій формі.

Вихідним параметром є час, протягом якого повинне горіти зелене світло для однієї із сторін.

Результати

Для **розв'язання задачі** за допомогою *теорії статистичних рішень* вхідні данні представляються у вигляді законів розподілу ймовірностей, які характеризують ймовірність появи певного числа ТЗ на перехресті за одиницю часу, тобто інтенсивність вхідного потоку.

Нехай N – число ТЗ, які прибули на перехрестя справа, а M – кількість ТЗ, що очікують на проїзд по центру. В обох випадках потоки ТЗ описуються законом розподілу Пуассона [6], тому що саме цей розподіл найчастіше зустрічається на практиці в подібних ситуаціях. Кількість ТЗ справа і в центрі є незалежними величинами.

Для прийняття рішень використаємо критерій Байеса, який можна записати у вигляді:

$$R(E) = \iint g(E, N, M) \cdot P(N, MQ) dN dM, \quad (1)$$

де $R(E)$ – середній ризик від прийняття рішення E ; $P(N, M)$ – ймовірність спільної появи конкретних значень величин N і M ; $g(E, N, M)$ – функція втрат, яка характеризує втрати від прийняття рішення E при заданих значеннях N і M .

Функція втрат враховує час простою ТЗ на перехресті, а також швидкість проїзду перехрестя.

Будемо вважати, що для проїзду перехрестя одному ТЗ потрібна одна секунда. Виходячи з цього, функція втрат має такий вигляд:

$$g = \frac{N}{E} \cdot \left(\frac{T-E}{E} \right) \cdot b + \frac{M}{T-E} \cdot \left(\frac{E}{T-E} \right) \cdot b, \quad (2)$$

де b – коефіцієнт пропорційності;

T – інтервал регулювання.

Рішення вважається оптимальним, якщо йому відповідає мінімальний ризик.

З точки зору *теорії масового обслуговування* перехрестя являє собою одноканальну систему масового обслуговування без відмов. Довжина черги є необмеженою. Вважаємо, що потоки є пуассонівськими, з середніми інтенсивностями λ_1, λ_2 .

Позначимо через T часовий інтервал регулювання. Тоді

$$t_1 + t_2 = T, \quad (3)$$

де t_1 – час, протягом якого горить зелене світло для одного потоку.

t_2 – час, протягом якого горить зелене світло для іншого потоку.

Величину інтервалу t_1 і t_2 можна визначити таким чином:

$$t_1 = \alpha \cdot T, \quad t_2 = (1 - \alpha) \cdot T, \quad (4)$$

де $\alpha \in (0; 1)$ – коефіцієнт поділу інтервалу.

Важливою характеристикою СМО є інтенсивність обслуговування μ , яка характеризує кількість обслужених заявок за одиницю часу. Якщо задати середній час обслуговування однієї заявки (час проїзду перехрестя транспортним засобом) Δt , то можна визначити інтенсивність обслуговування для кожного з потоків.

$$\mu_1 = \frac{t_1}{\Delta t} = \frac{\alpha \cdot T}{\Delta t}, \quad \mu_2 = \frac{t_2}{\Delta t} = \frac{(1 - \alpha) \cdot T}{\Delta t}, \quad (5)$$

Знаючи інтенсивності вхідних потоків та інтенсивності обслуговування визначимо загальні втрати як суму втрат кожного потоку. Втрати одного потоку дорівнюють добутку середнього часу очікування однієї заявки і середньої кількості заявок у черзі цього потоку. Таким чином загальні втрати обчислимо за формулою:

$$G = \frac{\rho_1^3}{\mu_1 \cdot (1 - \rho_1)^2} + \frac{\rho_2^3}{\mu_2 \cdot (1 - \rho_2)^2}, \quad (6)$$

де $\rho_1 = \frac{\lambda_1}{\mu_1}$, $\rho_2 = \frac{\lambda_2}{\mu_2}$ – інтенсивності завантаження системи від кожного потоку.

Таким чином, задача знаходження оптимального режиму регулювання перехрестя зводиться до мінімізації загальних втрат по α .

При використанні *нечіткої логіки* вхідні дані представляються у вигляді лінгвістичних змінних, кожній з яких відповідає кілька термів.

Вхідні лінгвістичні змінні приймають такі значення:

▪ “інтенсивність прибуття ТЗ, справа”

A = {“дуже мала”, “мала”, “середня”, “велика”, “дуже велика”};

▪ “інтенсивність прибуття ТЗ по центру”

Q = {“дуже мала”, “мала”, “середня”, “велика”, “дуже велика”};

Вихідний параметр “час до відключення зеленого сигналу світлофора” приймає такі значення

E = {“дуже малий”, “малий”, “середній”, “великий”, “дуже великий”}

Функції належностей вхідних величин мають форму, яка відповідає закону розподілу ймовірностей Пуассона. Функції належностей вихідної змінної мають форму, яка відповідає нормальному закону розподілу ймовірностей.

Вхідні величини змінюються в таких діапазонах:

- “інтенсивність прибуття ТЗ справа” [0; 10];
- “інтенсивність прибуття ТЗ по центру [0; 10];
- “Час до відключення зеленого сигналу світлофора” [0;10].

Нечіткий логічний висновок здійснюється за допомогою нечіткої бази знань, яка складається із сукупності правил виду “якщо–то”.

Дефазифікація нечіткої множини здійснюється методом центра тяжіння.

Результати тестування моделей наведені у таблиці 1.

Таблиця 1

Метод	Інтенсивності потоків			
	$\lambda_1 = 0,$	$\lambda_1 = 7,$	$\lambda_1 = 6,$	$\lambda_1 = 3,$
	$\lambda_2 = 7$	$\lambda_2 = 0$	$\lambda_2 = 6$	$\lambda_2 = 7$
ТМО	0,1Т	0,9Т	0,5Т	0,4 Т
ТСР	0,1Т	0,9Т	0,5Т	0,4Т
Нечітка логіка	0,101Т	0,87Т	0,5Т	0,292Т

Де T – інтервал регулювання.

Тестування моделей показало, що при однаковій інтенсивності з двох сторін усі методи дають однаковий результат. При різних початкових даних збігаються результати теорії статистичних розв'язків і масового обслуговування. Результат, отриманий за допомогою нечіткої логіки, відрізняється від решти. Слід зазначити, що теорія статистичних розв'язків не накладає обмежень на вигляд законів розподілу вхідних величин, а тому є більш універсальною, ніж теорія масового обслуговування.

Враховуючи проведені дослідження, пропонується метод прийняття рішень на основі представлення нечітких і стохастичних даних у вигляді узагальнювальних функцій (УФ) [1] і застосування алгоритму, аналогічного теорії статистичних розв'язків. Формальна система узагальнюючих функцій G складається з правил утворення формул, правил переходу від формальних систем чітких чисел R , випадкових величин P , нечітких чисел A до системи узагальнювальних функцій G і назад, системи аксіом та правил виведення. Для стохастичного даного УФ збігається за властивостями з щільністю розподілу ймовірностей. Для нечіткого даного УФ визначається нормованою за площею функцією належності.

При врахуванні стохастичної і нечіткої невизначеності формула (1) записується за допомогою узагальнюючих функцій наступним чином:

$$R(E) = \iint g(E, N, M) \cdot \frac{\beta(M/N)\beta(N)}{\int \beta(M/N)\beta(N)dN} dN dM \quad (7)$$

де $\beta(M/N)$, $\beta(N)$ – узагальнювальні функції.

В розподіленій системі загальний ризик є сумою ризиків у всіх точках керування

$$R(E) = \sum_i R(E_i).$$

Значення ризиків в окремих точках є залежними. Для підрахунку загального ризику введемо поняття “сусідньої точки”. Дві точки будемо вважати сусідніми, якщо інтервал керування T системою більший за час τ передавання впливу між цими точками. При керуванні транспортними потоками точками керування є перехрестя, час передавання впливу між сусідніми перехрестями дорівнює часу руху транспорту між цими перехрестями. Якщо обрати інтервал керування так, щоб у кожному напрямку від точки X_0 сусідньою була лише одна точка (для чотиристороннього перехрестя це всього чотири точки X_1 - X_4 , то УФ у виразі (7) можуть бути знайдені за допомогою рівняння лінійного прогнозу на інтервал τ

$$m_{N0/1} = m_{N0} + r_{01}(\tau) \sqrt{\frac{D_0}{D_1}} [N_1(t - \tau) + M_1(t - \tau) - m_{N1+M1}].$$

Висновки

Проведений аналіз методів прийняття рішень в умовах невизначеності показав, що для задач керування розподіленими системами перспективними є методи, що ґрунтуються на теорії статистичних рішень. Запропоновано методику прийняття рішень в умовах комбінованої невизначеності, яка дозволяє використовувати нечіткі і стохастичні дані і може використовуватись для розв'язання практичних задач керування розподіленими системами. Методика враховує взаємний вплив процесів у сусідніх точках розподіленої системи.

Список літератури

1. Дубовой В. М., Глонь О. В. Использование обобщенной вычислительной модели в интеллектуальных системах управления // Вісник Технологічного університету Поділля – 2002. – №3. – Т.1(41). – С.122–125
2. Мелихов А. Н., Берштейн Л. С., Коровин С.Я. Ситуационные советующие системы с нечеткой логикой. – М.: Наука. Гл. ред. физ.-мат. лит., 1990 – 272с.
3. Нечеткие множества в моделях управления и искусственного интеллекта / Под ред. Д.А. Поспелова. – М.: Наука. – 1986. –312с.
4. Репин В. Г., Тартаковский Г. П. Статистический синтез при априорной неопределенности и адаптация информационных систем. М.: Советское радио, 1977, 432с.
5. Савчук В.П. Байесовские методы статистического оценивания.– М.: Радио и связь, 1989.
6. Таха Х. Введение в исследование операций: В 2-х книгах, Кн. 2: Пер. с англ. – М.: Мир, 1985. – 496с.
7. С.Р. Pappis and E.H. Mamdani “A fuzzy logic controller for traffic junction”. – IEEE Trans. On Systems, Man and Cybernetics Vol. SMC-7, No 10, Oct 1977, pp. 707–717

Дубовой Володимир Михайлович, доктор технічних наук, професор, завідувач кафедри КСУ, Вінницький національний технічний університет, Хмельницьке шосе, 95, Вінниця, 21021, Україна, тел.: (0432) 44-01-57.

Ковалюк Олег Олександрович, студент, Вінницький національний технічний університет, Хмельницьке шосе, 95, Вінниця, 21021, Україна, тел.: (0432) 32-04-61.

УДК 681.3.07

Ю. В. Шабатура

РОЗПОДІЛЕНА ІНФОРМАЦІЙНА СИСТЕМА ПІДТРИМКИ КОЛЕКТИВНОГО ПОШУКУ ОПТИМАЛЬНИХ РІШЕНЬ З ІНТЕРВАЛЬНИМ УПРАВЛІННЯМ

Вінницький національний технічний університет, Вінниця

Вступ

Якщо розглядати діяльність людини саме як Homo Sapiens, тобто як розумної істоти, то очевидно, що однією з найяскравіших ознак, які відрізняють нас від інших представників живої природи є здатність людини до творчої діяльності і зокрема до вищої форми її реалізації у вигляді відкриттів та винаходів. Саме завдяки результатам такої діяльності людство отримало сучасну цивілізацію з її стрімкими переходами на нові, вищі рівні розвитку в багатьох сферах, які були, є і будуть неодмінно пов'язані з визначними відкриттями і винаходами. Тому перспективні наукові дослідження і винахідництво є пріоритетними напрямками, які підтримуються урядами в усіх країнах світу. Однак, сьогодні в епоху потужної інтеграції економіки виникає чимало проблем, які мають глобальний характер, і над вирішенням яких працюють як окремі вчені, так і цілі дослідницькі колективи у багатьох країнах світу. Зрозуміло, що якби вчені, які працюють над вирішенням однотипних задач, могли працювати в одному колективі, то час змарнований на пошуки найефективніших рішень, був би набагато коротшим. Історія розвитку науки наводить чимало прикладів [1,2], які підтверджують справедливості даного висновку. Тим прикріше, що на час здійснення багатьох з них вже були відомі основні методи та принципи ефективного об'єднання та інтенсифікації інтелектуальних зусиль [3] окремих вчених-дослідників і винахідників у пошуку рішень технічних задач на рівні винаходів. До найвідоміших таких методів прийнято відносити: метод мозкового штурму, евристичний алгоритм, метод параметричного синтезу, та інші.

Сьогодні, завдяки виникненню і розвитку глобальної світової мережі Internet, яка забезпечує абсолютно демократичні і унікальні можливості пошуку і обміну інформацією [4], створюється реальна основа для здійснення якісно нової, „розподіленої” технології пошуку рішень творчих задач. Суть цієї технології зводиться до створення динамічної, розподіленої інформаційної системи, яка дозволить інформаційно об'єднувати в реальному масштабі часу географічно віддалених один від одного винахідників та вчених, які займаються пошуками ефективних рішень творчих задач в окремо взятій предметній області.

Постановка задачі

Метою даної роботи є дослідження можливостей застосування Інтернет-технологій для здійснення інтенсифікації процесів пошуку рішень творчих задач винахідницького рівня, шляхом створення розподіленої динамічної інформаційної системи, яка забезпечить інформаційне об'єднання в реальному масштабі часу географічно віддалених один від одного дослідників. Функціональне управління цією інформаційною системою здійснюватиметься через виділення обмежених часових інтервалів для здійснення тих або інших функцій.

Граничні умови верхнього рівня цієї технології повинні відповідати семантичним обмеженням, тобто вона повинна залишатися не суперечливою в сенсі виконання в процесі вирішення задач творчого рівня чотирьох основних етапів:

1. Постановка задачі.
2. Пошук варіантів її вирішення.
3. Аналіз знайдених варіантів.
4. Оцінка варіантів і вибір оптимального.

Застосування сучасних методів творчого пошуку дозволяє раціоналізувати процес вирішення складних винахідницьких задач. Проте, в зв'язку з розвитком комп'ютерних мереж доцільно оцінити перспективи розробки та практичного використання методів творчих пошуків рішень винахідницьких задач в „розподіленій” – мережній версії.

Для досягнення поставленої мети в даній роботі пропонуються розв'язати такі задачі:

1. Розробити структуру системної конфігурації інформаційної системи для реалізації

- мереженого варіанту комплексного аналізу і методу мозкового штурму в застосуванні до пошуку рішень винахідницьких задач;
2. Розробити математичні моделі і критерії визначення ефективності для оцінки отриманих рішень.

Вступні зауваження

У формулюванні цілей даної статті є певні внутрішні протиріччя, які необхідно попередньо розглянути. Мова далі буде йти про технології пошуку рішень творчих задач. Справа в тому, що під технологією ми розуміємо чітко визначений, рутинний процес, виконання якого повинно невідворотно приводити до заданої цілі. На противагу цьому під творчістю ми розуміємо мистецтво, яке має не зовсім осмислений, невизначений характер, а тому, як правило, доступне лише окремим найталановитішим представникам людства. Разом з цим дослідження творчих процесів пошуків вдалих рішень технічних задач [3,5] дозволили визначити певні чітко структуровані закономірності, яким можна навчитися і використання яких дозволить гарантовано досягати потрібних результатів. До особливих ознак таких технологій потрібно віднести те, що їхнє застосування полегшує подолання психологічних бар'єрів, які неодмінно виникають на шляху пошуків по справжньому революційних нових рішень.

Синтез структури узагальненого алгоритму пошуку нових технічних рішень

Створення інформаційної системи на основі використання засобів інформаційної техніки та мережових технологій вже в силу природи своєї бази передбачає наявність певних алгоритмів. Однак в даному випадку функціонування системи призначено для виконання допоміжних, свого роду сервісних функцій, які сприятимуть і полегшуватимуть пошук прийнятних рішень з боку людей – учасників процесу. Тому при цьому теж необхідний алгоритм, однак вимоги до його структуризації можуть бути дещо послабленими, тобто мова може йти про евристичні алгоритми.

Подібні алгоритми включають в себе систему правил та інформаційну базу. Таким чином використання Інтернет-технологій дозволить збільшити ефективність подібного комплексного методу пошуку нових технічних рішень за рахунок створення можливості одночасного віддаленого доступу множини користувачів – учасників процесу, до єдиної інформаційної бази. А якщо остання буде мати можливість вільної зміни інформаційного ресурсу, за умови обов'язкового збереження початкового та всіх наступних постановок та варіантів вирішень конкретної задачі (інформаційні аналоги в Інтернет: гостьові книги; форуми; конференції і т.д.) то, тоді на сервері буде функціонувати динамічно оновлювана інформаційна база, яка в нашому випадку для користувачів буде представлена як блок перетворення технічної системи.

На рис. 1 представлена структура, яка відображає загальнотехнічну методику пошуку нових рішень, що орієнтована на використання комп'ютерних інформаційних баз. Ядром, що об'єднує евристичний алгоритм, є постановка та уточнення задачі. Це ядро є відкритим для зовнішніх учасників і воно повинно відображати еволюційний процес змін в тезаурусі постановки задачі. В операційній частині передбачається здійснення синтезу системи та перевірка його автентичності і визначення складу синтезованих підсистем. На всіх етапах учасники процесу мають вільний доступ до інформаційної бази, в якій відображається синтез можливих варіантів рішень та перевірка правильності синтезованих рішень. Завершальними етапами алгоритму є оцінка та вибір рішень з наступним переглядом можливості їх розвитку. Якщо можливості розвитку будуть вичерпані, то буде прийнято остаточний варіант рішення.

Застосування Інтернет-технологій в режимі on-line дозволить кожному з учасників такого колективного дистанційного пошуку ефективного вирішення творчої технічної задачі зможе не тільки пасивно спостерігати, але і активно впливати на всі етапи пошуку, синтезу і перевірки правильності синтезованих рішень незалежно від їх географічного розташування, тобто вони стають дистанційними операторами розглянутого евристичного алгоритму.

Недолік розглянутого підходу полягає в тому, що для успішного пошуку і виділення найефективнішого, оптимального рішення тут потрібна участь здібного адміністратора інформаційної бази, на якому лежить відповідальність за успішний вибір і правильний розвиток рішення, а тому вплив суб'єктивного підходу в даному випадку буде неминучим.

Синтез інформаційної системи реалізації Brain-Storming в Інтернет-технології

На мій погляд, найперспективнішим методом інтенсифікації процесу рішення винахідницьких задач з використанням Інтернет-технологій є метод мозкового штурму (Brain-Storming). Даний метод був розроблений в США. В наукових кругах він став відомим після виходу в 1953 році книги А. Осборна „Керована уява” [6], у якій були розкриті загальні принципи і процедури творчого мислення.

Структурно метод достатньо простий. Він являє собою двоетапну процедуру пошуку розв'язку задач. На першому етапі висувуються ідеї, на другому вони аналізуються, конкретизуються і

розвиваються. Діяльність учасників в рамках цих етапів повинна підпорядковуватися певним правилам. На етапі висунення (генерації) ідей їх три:

1. Заборона будь-якої критики висунутих ідей.
2. Заборона обґрунтування ідей.
3. Заохочення всіх запропонованих ідей, в тому числі і абсолютно нереальних та фантастичних.

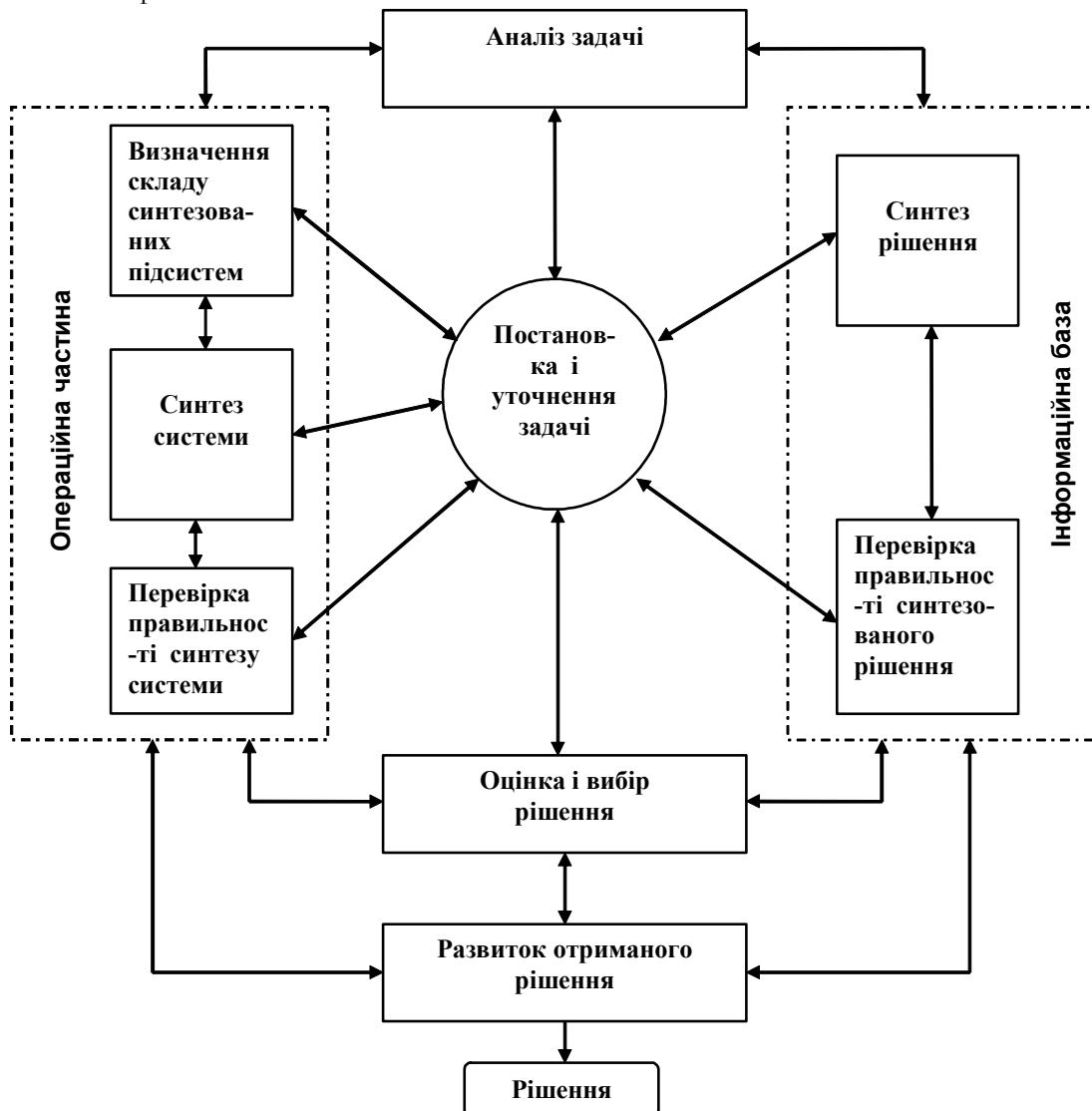


Рис. 1. Структура евристичного алгоритму пошуку нових технічних вирішень з застосуванням комп'ютерної інформаційної бази

На етапі аналізу потрібно дотримуватися тільки одного основного правила – виявляти раціональну основу в кожній ідеї, що піддається аналізу.

Вказані положення якнайкраще виконуються саме при застосуванні Інтернет-технологій. Адже саме у цьому випадку учасники мозкового штурму знаходяться не в одному приміщенні, а можуть бути віддалені один від одного на тисячі кілометрів, тому будь-який вплив, який може вплинути на висунення нових ідей на першому етапі тут абсолютно виключений. Під час реалізації другого етапу, коли масив згенерованих ідей буде ретельно аналізуватися іншими учасниками процесу (аналітиками), учасники першого етапу можуть залишатися пасивними спостерігачами. Якщо протягом часу, відведеного на здійснення другого етапу, в них виникнуть нові ідеї розвитку запропонованих рішень, то вони зможуть їх представити після закінчення аналізу на повторному етапі генерації додаткових ідей. Схема взаємодії учасників процесу, інформаційної бази і допоміжних програм зображена на рисунку 2.

В даній схемі важлива роль належить редактору процесу мозкового штурму (User Editor). Він повинен повідомити учасників процесу про дату і час початку чергового сеансу. Надати їм паролі для доступу та іншу необхідну сервісну інформацію. В його обов'язки також входить формулювання

постановки задачі в текстовому, графічному, вербальному та інших формах представлення. Причому в цьому питанні важливим є лише максимально повне розкриття суті задачі і представлення цієї інформації на сторінці сайту – „Постановка задачі”. Як і адміністратор, редактор процесу повідомляє про тривалість часових інтервалів, які виділяються на здійснення етапів процесу і попереджує про моменти їх початку та закінчення. Отже єдиним ресурсом управління з боку редактора в цій технології є управління ресурсами часу – інтервальне управління [7].

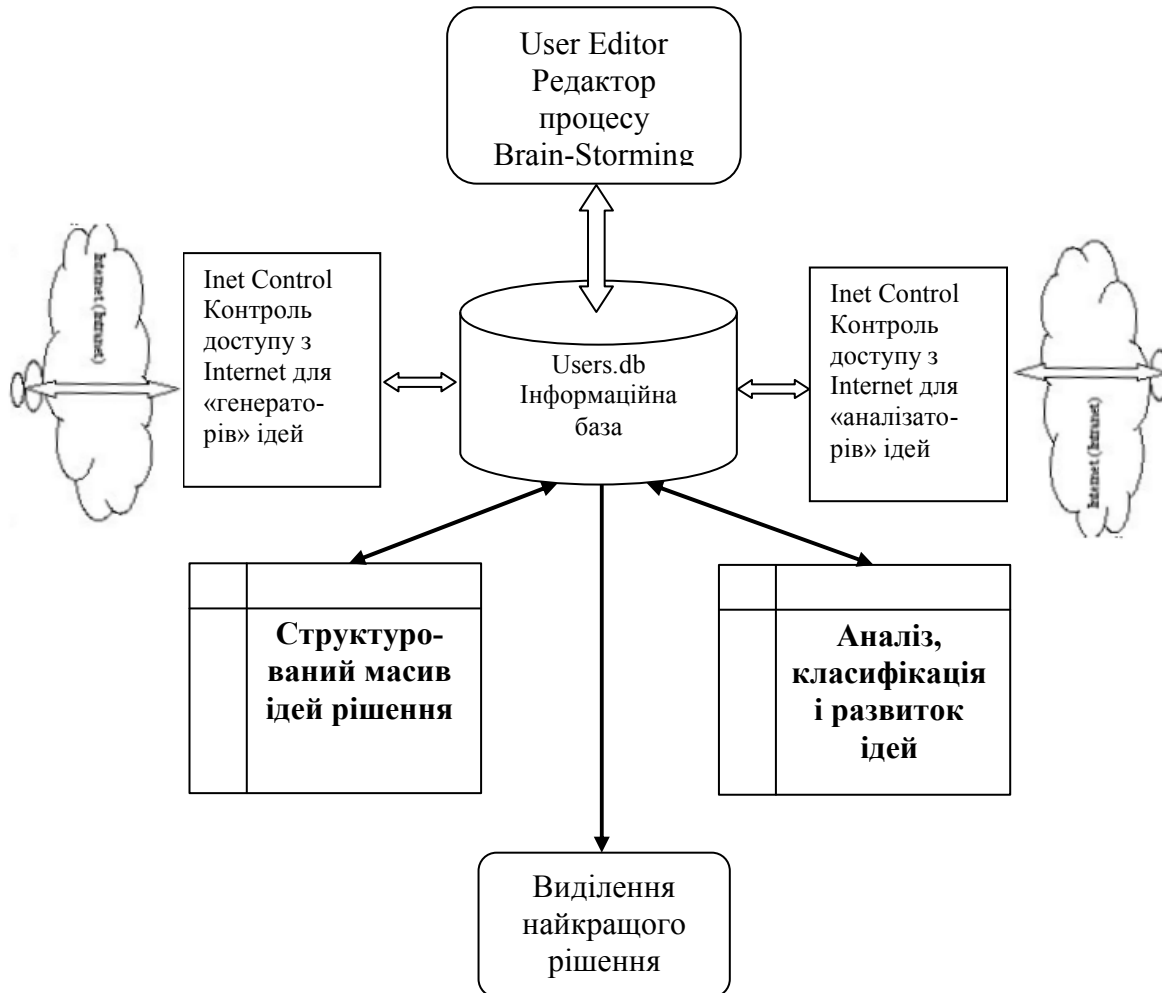


Рис. 2. Схема взаємодії об'єктів та суб'єктів інформаційної системи для реалізації Brain-Storming з підтримкою Інтернет-технології

Управління режимами доступу для учасників, які генеруватимуть ідеї, та для учасників, які будуть аналізувати масив отриманих ідей, буде здійснюватися за допомогою модифікованих програм контролю доступу до інформаційної бази з Інтернету (Inet Control).

В динаміці Brain-Storming в Інтернет-технології буде здійснюватися таким чином. В обумовлений час учасники Brain-Storming, які входять в групу генераторів, отримують доступ до сторінки сайту з початковим формулюванням задачі. Через певний часовий інтервал, який виділяється для ознайомлення з завданням, вони отримують повномасштабний доступ до індивідуальних сторінок, на яких вони можуть формулювати будь-які варіанти рішень, причому кожний з них може ознайомитися з вмістом сторінок інших учасників, однак при цьому він не матиме можливості зробити там жодних змін.

Після закінчення часового інтервалу, який виділяється для проведення першого етапу процесу, повномасштабний доступ учасників першої групи до своїх індивідуальних сторінок закінчується. Тепер доступ до отриманого масиву рішень отримує друга група учасників процесу, які виконують аналіз, розвиток і вибір найкращого варіанту рішення запропонованої задачі. Виділене таким чином рішення виставляється на спеціальній сторінці для загального обговорення з метою виділення його сильних та слабких сторін, та отримання нових пропозицій направлених на його покращення.

Математичні моделі та критерії оцінки ефективності отриманих рішень

Для оцінки ефективності отриманих рішень пропонується використовувати формальний апарат математичних моделей, які відображають зв'язок між показниками початкової технічної системи та параметрами її елементів і таку ж залежність, тільки отриману для нової технічної системи, яка представлятиме собою оптимальне рішення задачі.

В якості прикладу розглянемо деякий найбільш важливий показник системи M_i , та визначимо його залежність:

$$M_i = f_i([P_n]_i), \quad (1)$$

де f_i – функціональна залежність показника M_i від параметрів елементів системи $(P_n)_i = P_1, \dots, P_k$

Якщо параметри елементів самі є залежними, то потрібно визначити і їх функції залежності.

$$\begin{aligned} P_1 &= f_1((P_n)_{i1}) \\ &\dots \dots \dots \\ P_j &= f_j((P_n)_{ij}) \end{aligned} \quad (2)$$

Таким чином модель буде редукована до рівня незалежних параметрів. У цьому випадку модель (1) можна подати у вигляді:

$$M_i = f_i((P_n)_{i1}, \dots, (P_n)_{ij}, P_{j+1}, \dots, P_k) \quad (3)$$

В якості критерію ефективності для порівняння отриманих технічних рішень можна використовувати квадратичну залежність виду:

$$K_E = \sum_{i=1}^n \frac{M_i^2}{C_i}, \quad (4)$$

де C_i – загальна вартість забезпечення рівня значення параметра M_i .

Найбільш прийнятним буде вважатися те рішення для якого даний критерій буде приймати максимальне значення.

Звичайно для повноцінного оцінювання отриманих технічних рішень недостатньо здійснювати порівняння по одному, навіть досить важливому критерію. Для цього необхідно розрахувати значення критеріїв ефективності по всіх основних параметрах, а після цього виконати рандомізацію по рівню значення цих параметрів, скласти з них матриці, та виконати порівняння норм отриманих матриць.

Висновки

В даній роботі виконані дослідження можливості використання Інтернет-технологій для створення розподіленої динамічної інформаційної системи, яка забезпечить інформаційне об'єднання в реальному масштабі часу географічно віддалених один від одного дослідників з метою інтенсифікації процесів пошуків рішень технічних задач винахідницького рівня. Представлені структури системних конфігурацій та інформаційних взаємодій для реалізації розподілених варіантів комплексного аналізу і методу мозкового штурму. Виділені особливості, які дозволяють покращити ефективність мозкового штурму при застосуванні Інтернет-технологій. Розроблені математичні моделі та критерії, які дозволяють виконати об'єктивну оцінку ефективності отриманих рішень.

Список літератури

1. Гвишиани Д. М., Кедров Б. М., Микулинский С. Р. и др. Очерки истории и теории развития науки. М.: Наука, 1969. – 423 с.
2. Пономарев Л. И. Под знаком кванта – М.: Наука, 1989. – 368 с.
3. Альтшуллер Г.С. Найти идею. Введение в теорию решения изобретательских задач. Новосибирск.: Наука, 1986. – 176 с.
4. Когаловский М. Р. Перспективные технологии информационных систем. – М.: ДМК Пресс; – М.: Компания АйТи, 2003. – 288 с.
5. Кудрявцев А. В. Методы интуитивного поиска технических решений. – М.: Речной транспорт, 1991. – 111с.
6. Osborn A.F. Applied imagination. New-York. 1953. – 76 p.
7. Шабатура Ю. В. Основи теорії та практики інтервальних вимірювань: Монографія. – Вінниця: Універсум-Вінниця, 2003. – 167 с.

Шабатура Юрій Васильович, кандидат технічних наук, доцент, кафедра метрології та промислової автоматики, Вінницький національний технічний університет, Хмельницьке шосе 95, м. Вінниця, Україна. 21021. тел. 8(0432)44-05-71 E-mail: shabatura@vstu.vinnica.ua

УДК 519.6:517.44

В. М. Дубовой, К. В. Казиміров, О. В. Олійник

ЗАСТОСУВАННЯ ІНТЕРВАЛЬНОГО ПІДХОДУ ДО ОПЕРАТОРНИХ МОДЕЛЕЙ СИСТЕМ КЕРУВАННЯ ЗА УМОВ НЕВИЗНАЧЕНОСТІ

Вінницький національний технічний університет, Вінниця

Вступ

Переважає більшість інформаційних систем працює в умовах невизначеності різної природи. Джерелом невизначеності може бути недостатньо повне знання предметної області, недостатня інформація про конкретну ситуацію, недостатність достовірної інформації про значення параметра системи, яка може бути викликана багатьма причинами, недостатність достовірної інформації про значення даного, яка може бути зумовлена багатьма причинами. Ступінь неповноти інформації зумовлена способами отримання інформації та її достовірності. Врахування невизначеності впливає на методологію проектування та оптимізації систем управління. Тому **проблема** моделювання та аналізу систем управління в умовах невизначеності є **актуальною**.

На сьогоднішній день існує декілька підходів до розв'язання проблеми невизначеності при моделюванні систем управління. Слід відзначити історично перший підхід до врахування невизначеності – врахування невизначеності стохастичними методами, тобто з застосуванням апарата теорії імовірності [1]. В цій групі методів невизначена величина розглядалася як випадкова величина з відповідними характеристиками. Система управління розглядалася як сукупність блоків, які здійснюють відповідні перетворення над характеристиками цих випадкових величин [2]. Але надмірна складність математичних характеристик перетворювачів призвела до поступового переходу до інших методів врахування невизначеності.

Паралельно із стохастичними методами як їх альтернатива розвинулася теорія нечітких множин, яка замість імовірності пропонувала розглядати ступінь належності [3]. Ця теорія дозволяла найповніше враховувати невизначеність експертних оцінок.

В [6] викладений метод моделювання широкого класу інформаційних систем в умовах комбінованої стохастичної та нечіткої невизначеності. Метод ґрунтується на перетворенні характеристик невизначеності (щільності розподілу ймовірності стохастичних даних $f(x)$ і функції належності нечітких даних $\mu(x)$) на узагальнювальні функції $\beta(x)$ і моделювання їх перетворень за допомогою інтегральних операторів виду

$$\beta(y) = \int_{\Omega_X} n \int \beta(x) \phi(x, y) dx, \quad (1)$$

де $\phi(x, y)$ – ядро перетворення, яке залежить від характеристик системи, що моделюється.

Кратність інтегрування n в операторі (1) визначається розмірністю вектора вхідних даних та складністю перетворення. Так, наприклад, при моделюванні нелінійного унарного статичного перетворення кратність інтегрування 1, лінійного бінарного статичного перетворення – 2, нелінійного бінарного статичного перетворення – 3 (воно представляється композицією лінійного бінарного і нелінійного унарного перетворень), лінійного динамічного перетворення – n (воно представляється композицією n лінійних бінарних перетворень, де n залежить від порядку передаточної функції перетворення). Для реальної системи кратність інтегрування може досягати десятків, що при чисельній реалізації навіть на комп'ютері з високою швидкістю займає досить багато часу.

Таким чином можна сформулювати **задачу** підвищення швидкості моделювання операторним методом.

Результати

При стохастичному представленні невизначеної інформації існує представлення закону розподілу в інтервальному вигляді за допомогою функцій від імовірності потрапляння випадкової величини в інтервал (рис.1,а), тобто представлення функції розподілу через сукупність довірчих інтервалів [4]. При нечіткому представленні невизначеної інформації існує можливість представлення функції

приналежності нечіткого числа у α -рівневому вигляді (рис.1,б) за допомогою LR-форми [3]. Таким чином є можливість представити нечітку і стохастичну невизначеність за допомогою інтервального числення [5].

Оскільки під час розв'язання запропонованої задачі виникає необхідність поєднання стохастичної та нечіткої невизначеностей в інформації, то для розв'язання цієї проблеми пропонується застосувати метод узагальнювальних функцій (УФ) [6], який дозволяє поєднувати два види невизначеностей в межах однієї моделі. До того ж однією з можливих форм запису УФ є запис за допомогою довірчих інтервалів.

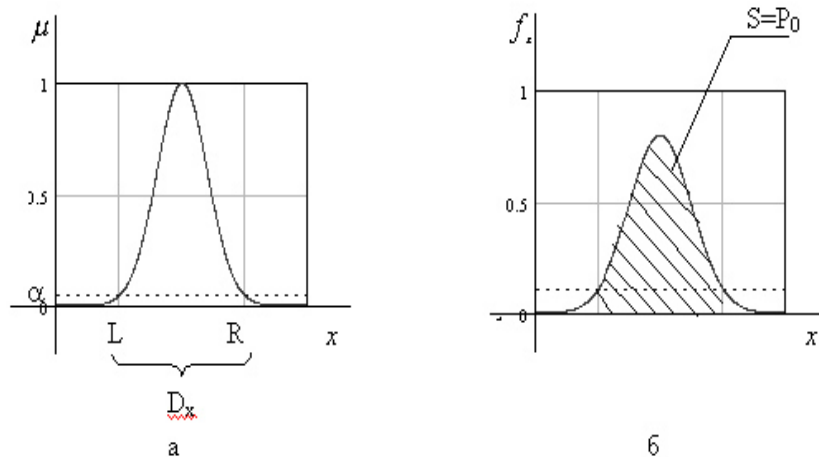


Рис. 1. Різні форми невизначеностей в інтервальному вигляді

а) LR-форма на α -рівні; б) довірчий інтервал

Будь-яку систему управління можна представити за допомогою комбінації таких перетворювачів [2]:

- лінійні статичні перетворювачі;
- нелінійні статичні перетворювачі однієї невизначеної величини (логарифмування, піднесення до степеня тощо);
- нелінійні бінарні перетворювачі (додавання, множення тощо);
- лінійні динамічні перетворювачі (інтегрування за часом, диференціювання за часом тощо).

Наявний математичний апарат інтервального аналізу [5] дозволяє виконати вищевказані перетворення, за умови отримання результуючого інтервалу з характеристиками, які відповідають характеристикам початкових інтервалів та узагальнювальних функцій.

На першому етапі отримані результати для випадків лінійних та нелінійних перетворювачів одного аргументу.

Нехай R – множина всіх дійсних чисел. Тоді під *інтервалом* $[a, b]$, $a \leq b$, якщо не обумовлено іншого, будемо розуміти замкнуту обмежену підмножину R виду:

$$[a, b] = \{x \mid x \in R \wedge a \leq x \leq b\}, \quad (2)$$

Множину всіх інтервалів позначимо через $\mathbf{I}(R)$. Елементи множини $\mathbf{I}(R)$ будемо позначати прописними літерами. Якщо X – елемент $\mathbf{I}(R)$, $X \in I(R)$, то його правий і лівий кінці будемо позначати як \underline{x} ; \bar{x} : $X = [\underline{x}; \bar{x}]$. Елементи $\mathbf{I}(R)$ інтервальні числа.

Символи \in, \cap, \cup тощо розуміються в звичайному теоретико-множинному розумінні, причому \subset позначає не обов'язково строге включення, тобто співвідношення $A \subset B$ допускає рівність інтервалів. Два інтервала A і B *рівні* тоді і тільки тоді, коли $\underline{a} = \underline{b}$, $\bar{a} = \bar{b}$.

Вироджений інтервал, тобто інтервал зі збіжними кінцями $\underline{a} = \bar{a} = a$, отожденимо з дійсним числом a . Таким чином, $R \in I(R)$.

При проведенні інтервального моделювання всі операції необхідно проводити в рамках інтервальної арифметики. Класична інтервальна арифметика - це алгебраїчна система $\langle \mathbb{R}, +, -, \cdot, / \rangle$, носій якої \mathbb{IR} утворений інтервалами дійсної осі $\mathbf{R}[\underline{x}, \bar{x}]$, $\underline{x} \leq \bar{x}$, а бінарні операції додавання, віднімання, множення і ділення визначені таким чином, що фундаментальна властивість

$$x * y = \{x * y \mid x \in x, y \in y\}, \quad (3)$$

повинна виконуватись для всіх інтервалів x, y , таких що $(x * y), * \in \{+, -, \cdot, /\}$, вірно для всіх $x \in x, y \in y$.

Відповідно до цього можна визначити основні арифметичні операції з інтервалами. За допомогою основних алгебраїчних операцій та властивостей інтервального обчислення можливо отримати набір функцій для інтервальних чисел. Якщо ще врахувати ряд співвідношень між окремими функціями, то стає можливим отримати необхідні вирази для реалізації майже всіх елементарних функцій. [5]

Подання будь-якої загальної функції g^R у вигляді інтервальної функції g^{IR} є однією з найважливіших проблем інтервального аналізу.

Інтервальним розширенням функції $g^R(x)$, $x \in \mathbb{R}$, називають такий елемент $g^{IR}(\mathbf{X})$, $\mathbf{X} \in \mathbb{IR}$, що при $x \in X$ виконується умова $g^R(x) \in g^{IR}(\mathbf{X})$.

Інтервальне розширення позначається як:

$$Di_{x \rightarrow X} g^R(x) = g^{IR}(\mathbf{X}), \quad (4)$$

Функція $g^{IR}(\mathbf{X})$, що отримана заміною дійсного аргументу x в раціональній функції $g^R(x)$ інтервальним аргументом X з переходом в інтервальну арифметику, називається природним інтервальним розширенням.

Шляхом оператора Di можна перейти до дій над інтервалами.

Зворотною операцією щодо інтервального розширення є інтервальне звуження.

Застосування інтервальних засобів припускає побудову інтервальних розширень, на які накладаються умови найбільшої звуженості.

Розглянемо застосування інтервальних операцій до моделювання перетворень узагальнювальних функцій невизначеності в нелінійних унарних перетвореннях.

Відповідно до [6] для оцінки узагальнювальної функції β_y результату нелінійного перетворення $y = N(x)$ вхідного сигналу X з узагальнювальною функцією β_x потрібно застосовувати таку формулу:

$$\beta_y = N^{-1}(y) \cdot \beta_x [N^{-1}(y)], \quad (5)$$

Але поступовий обрахунок всіх функцій, які входять до формули (5), є складним та довгим процесом за будь-якої більш-менш складної функції $N(x)$. Застосуємо для їх перетворення інтервальну математику, що дозволить значно спростити і пришвидшити процес отримання результату.

Перетворення здійснюється у такій послідовності:

- На узагальнювальній функції вхідного сигналу β_x на потрібному рівні отримується інтервал D_x , який задовольняє умову $\int_{D_x} \beta_x(x) dx = C$.
- За допомогою природного інтервального розширення з функції $N(x)$ отримується її інтервальний відповідник: $N^{IR}(X)$.
- Використовуємо отриманий інтервал D_x як аргумент інтервальної функції $N^{IR}(X)$.
- Як результат отримуємо інтервал D_y .

Приклади застосування запропонованої методики наведені на рис. 2. Легко довести, що отриманий таким способом інтервал буде задовольняти основну умову

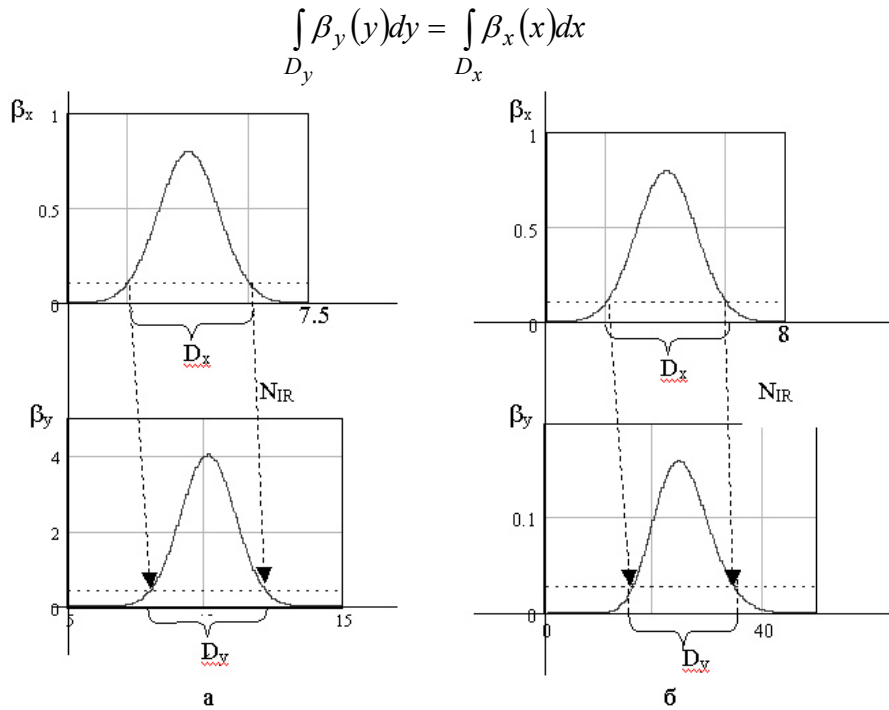


Рис. 2. Лінійне і нелінійне перетворення меж інтервалу

а) лінійне : $N(x)=2x \quad D_x : [4;6] \xrightarrow{N_{IR}} D_y : [4;12]$

б) нелінійне : $N(x)=x^2 \quad D_x : [4;6] \xrightarrow{N_{IR}} D_y : [16;36]$

Важливою умовою застосування інтервального підходу є однозначність представлення УФ у інтервальній формі. На рис. 3 показані приклади УФ, для яких можливе (а) і неможливе (б) інтервальне

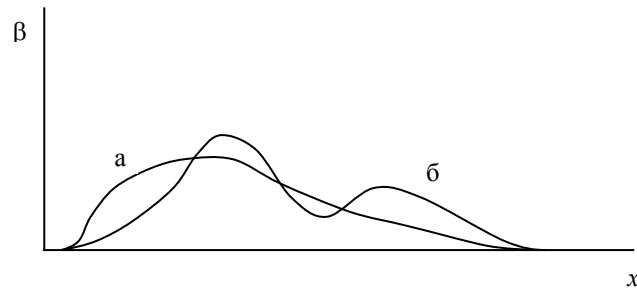


Рис. 3. Приклади узагальнювальних функцій

представлення. Очевидно, що для забезпечення однозначності представлення повинна виконуватися умова унімодальності. Але для виконання інтегрування в операторі (1) при інтервальному представленні унімодальною повинна бути не тільки УФ, але й добуток

$$F(x,y) = \beta(x) \varphi(x,y). \tag{6}$$

Проаналізуємо необхідні умови унімодальності функції $F(x,y)$. Можливі варіанти унімодальних функцій показані на рис.4.

Очевидно, функція $F(x,y)$ буде унімодальною, якщо на області визначення вона має три інтервали монотонності: інтервал монотонного зростання, інтервал стаціонарності і інтервал монотонного спадання. У загальному випадку кожен з інтервалів може містити лише одну точку.

Якщо функція диференційована на інтервалі, то її монотонність означає постійність знаку першої похідної

$$F'(x, y) = \beta'(x)\phi(x, y) + \beta(x)\phi'(x, y) \quad (7)$$

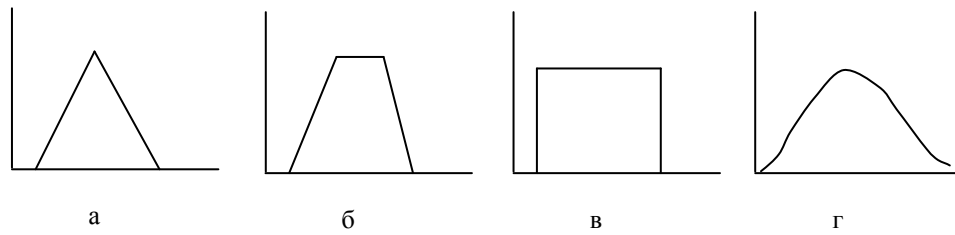


Рис. 4. Унімодальні функції

На інтервалі зростання $F'(x, y) > 0$, на інтервалі стаціонарності $F'(x, y) = 0$, на інтервалі спадання $F'(x, y) < 0$. Розглянемо розв'язок нерівності

$$\beta'(x)\phi(x, y) + \beta(x)\phi'(x, y) \lessgtr 0 \quad (8)$$

де знак \lessgtr означає один з символів відношення рівності/нерівності. Враховуючи, що УФ і ядро перетворення є додатно визначені, нерівність (4) може бути представлена у вигляді

$$\frac{\phi'(x, y)}{\phi(x, y)} \lessgtr -\frac{\beta'(x)}{\beta(x)} \quad (9)$$

З нерівності (9) видно, що у випадках, коли інтервали монотонності УФ і ядра перетворення збігаються, нерівність виконується, оскільки на інтервалі зростання $\phi'(x, y) > 0$, $-\beta'(x) < 0$, отже $\frac{\phi'(x, y)}{\phi(x, y)} > -\frac{\beta'(x)}{\beta(x)}$. На інтервалі спадання $\phi'(x, y) < 0$, $-\beta'(x) > 0$, отже $\frac{\phi'(x, y)}{\phi(x, y)} < -\frac{\beta'(x)}{\beta(x)}$. На

інтервалі стаціонарності $\frac{\phi'(x, y)}{\phi(x, y)} = -\frac{\beta'(x)}{\beta(x)} = 0$. У випадках, коли інтервал зростання ядра збігається з інтервалом стаціонарності УФ або інтервал стаціонарності ядра збігається з інтервалом спадання УФ нерівність (9) теж виконується. Для решти випадків умова (9) не виконується, отже функція $F(x, y)$ не є унімодальною.

Оскільки ядро $\phi(x, y)$ задається з параметром y , який впливає на зсув ядра вздовж вісі Ox , то всю область значень параметра Ω_y можна розділити на дві підобласті: Ω_y^1 , в якій умова унімодальності виконується, і Ω_y^2 , в якій умова унімодальності не виконується. Відповідно, в першій підобласті можливий прискорений інтервальный спосіб інтегрування в перетворенні (1), а в другій підобласті необхідно застосовувати один з класичних чисельних методів інтегрування.

Висновки

Запропонована методика отримання довірчих інтервалів після функціонального перетворення вхідної величини та виконання інтегральних перетворень дозволяє значно пришвидшити та спростити процес моделювання та аналізу систем управління в умовах комбінованої стохастичної та нечіткої невизначеностей. Подальший розвиток отриманих результатів дозволить спростити чисельні алгоритми виконання всіх операцій над узагальнювальними функціями.

Список літератури

1. Солодовников В.В. Статистическая динамика линейных систем автоматического управления. – М.: Физматгиз, 1960.
2. Анализ измерительных информационных систем. Маликов В.Т., Дубовой В.М., Кветный Р.Н., Исмагуллаев П.Р. – Ташкент: Фан, 1984. – 176 с.

3. Нечеткие множества в моделях управления и искусственного интеллекта / Под ред. Поспелова Д.А. – М.: Наука, 1986. – 312с.
4. Колемаев В.А. и др. Теория вероятности и математическая статистика. – М.: Высшая школа, 1991. – 400 с.
5. Шокин Ю.И. Интервальный анализ. - Новосибирск: Наука, 1981. - 112с.
6. Дубовой В.М., Глонь О.В. Моделювання систем керування в умовах невизначеності. – В.: УНІВЕРСУМ-Вінниця, 2004. – 170 с.

Дубовой Володимир Михайлович, доктор технічних наук, професор, завідувач кафедри КСУ, Вінницький національний технічний університет, Хмельницьке шосе, 95, Вінниця, 21021, Україна, тел.: (0432) 440157, E-Mail: dub@faksu.vstu.vinnica.ua

Казиміров Костянтин Володимирович, студент, Вінницький національний технічний університет, Хмельницьке шосе, 95, Вінниця, 21021, Україна, тел.: (0432) 32-04-61, E-Mail: kons_kaz@ukr.net

Олійник Ольга Вікторівна, студентка, Вінницький національний технічний університет, Хмельницьке шосе, 95, Вінниця, 21021, Україна, тел.: (0432) 46-07-72, E-Mail: oliynykolya@bigmir.net

УДК 621.396.2

Г. Л. Лисенко, І. І. Тютюнник

СИСТЕМА МОНІТОРИНГУ ОСТАННЬОЇ МИЛІ В ОПТИЧНИХ МЕРЕЖАХ

Вінницький національний технічний університет, Вінниця

Вступ

Для провайдерів оптичних мереж дуже важливо гарантувати користувачам надійність і безпеку передачі даних.

Існує багато різних систем моніторингу, кожна з яких має свої унікальні переваги, але майже всі з них достатньо складні і дорогі [1,2,3].

Тому створення системи, що є простою та недорогою з аналогічними параметрами є актуальним.

Будова системи

Пропонується система для моніторингу оптичних мереж з використанням часового ущільнення каналів (TDM) з фіксованими тайм-слотами. В залежності від протяжності лінії останньої милі можливі два варіанти будови системи. Перший варіант (більш простий і дешевий) призначений для мереж з максимальною довжиною волокна до 100м. Це пов'язано з тривалістю тайм-слотів в сучасних магістралях, яка становить 1мкс. За цей час світло у волокні проходить приблизно 200 м, але оскільки використовується рефлектометричне дослідження волокна, то це відповідає максимальній довжині 100 м. Другий варіант (більш складний, будується на базі першого) призначений для мереж з максимальною довжиною волокна більшою за 100 м.

Розглянемо спочатку будову першого (простого) варіанта системи.

Система складається з модуля керування, ОЧІР (оптичного часово-імпульсного рефлектометра), двох оптичних перемикачів та персонального комп'ютера (рис. 1).

Модуль керування координує роботу всієї системи. Він підключений до ОЧІР і до обох оптичних перемикачів. Він має інтерфейс з комп'ютером, який дозволяє керувати роботою системи. Також він підключений до мультиплексора та демультиплексора для синхронізації з тайм-слотами.

Оптичний перемикач №1 підключений до модуля керування і до оптичного перемикача №2. Він перемикає оптичні волокна, які йдуть до чи від користувачів до магістралі або до оптичного перемикача №2.

Оптичний перемикач №2 підключений до модуля керування, до оптичного перемикача №1 і до ОЧІР. Він підключає ОЧІР до входу (виходу) оптичного перемикача №1, який підключений до волокна, що йде до чи від користувача.

ОЧІР може працювати у двох режимах: моніторингу потужності, коли у волокно посилається лише один імпульс, а середня потужність сигналу що повернувся передається модулю керування, і в режимі моніторингу волокна, коли у волокно посилається лише один імпульс, а повна рефлектограма сигналу що повернувся передається модулю керування. Модуль керування запам'ятовує усі дані, які отримані від ОЧІР у вбудовану пам'ять.

Персональний комп'ютер під'єднаний до модуля керування. Він дозволяє налаштувати та управляти системою, переглядати стан системи, переглядати рефлектограми.

Другий варіант (складний) системи складається з модуля керування, ОЧІР, чотирьох оптичних перемикачів, мультиплексора хвильового ущільнення та персонального комп'ютера (рис. 2).

Модуль керування, ОЧІР та персональний комп'ютер виконують тут ті ж функції, що і у попередньому варіанті системи.

Оптичний перемикач №3 підключений до модуля керування, оптичного перемикача №4 і до оптичного перемикача №1. Він перемикає оптичні волокна, які йдуть до чи від магістралі, або до оптичного перемикача №4, або до оптичного перемикача №1.

Оптичний перемикач №4 підключений до модуля керування, оптичного перемикача №3 і до мультиплексора хвильового ущільнення. Він перемикає одне з оптичних волокон, які йдуть від оптичного перемикача №3, на мультиплексор хвильового ущільнення.

Мультиплексор хвильового ущільнення служить для об'єднання сигналів на різних довжинах хвилі, що подаються на нього від оптичного перемикача №3 та ОЧІР.

Оптичний перемикач №2 підключений до модуля керування, до оптичного перемикача №1 і до мультиплектора хвильового ущільнення. Він підключає вихід мультиплектора хвильового ущільнення до входу (виходу) оптичного перемикача №1, який підключений до волокна, що йде до чи від користувача.

Оптичний перемикач №1 підключений до модуля керування, оптичного перемикача №3 і до оптичного перемикача №2. Він перемикає оптичні волокна, які йдуть до чи від користувачів, до мультиплектора хвильового ущільнення або до оптичного перемикача №2.

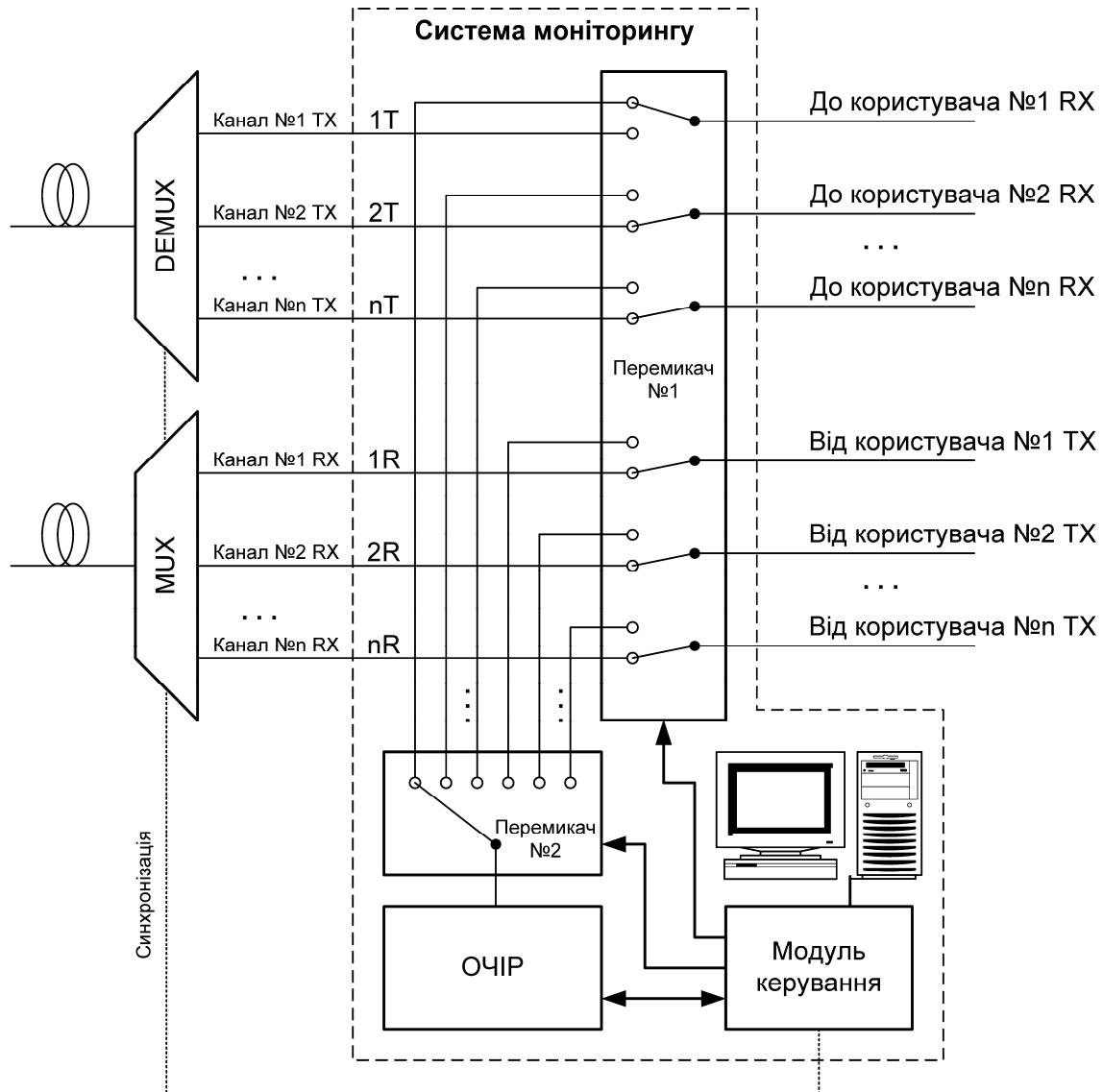


Рис. 1. Система моніторингу (перший варіант)

Робота системи

Розглянемо спочатку роботу першого (простого) варіанта системи.

В повнодуплексній оптичній мережі з часовим ущільненням каналів, коли магістраль розгалужується на волокна останньої милі (через демультиплексор та мультиплексор), в будь-який момент часу тільки по двох волокнах останньої милі будуть передаватись дані (перше – TX: від магістралі, друге – RX: до магістралі) [4]. За кожним волокном останньої милі зафіксований конкретний тайм-слот. Всі інші волокна останньої милі “темні” у цей час. Тому вони можуть бути легко перевірені. Але, як уже згадувалось вище, тайм-слоти досить коротко-тривалі, що обмежує максимальну протяжність тестування волокон.

Коли волокно перевіряється, воно підключається через два оптичних перемикачі (№1 та №2) до ОЧІР. Кожна перевірка вирівнюється в часі з початком тайм-слоту модулем керування.

ОЧІР посиляє сигнал у волокно і отримує відбитий сигнал [5]. Він вираховує середню потужність відбитого сигналу і посиляє його модулю керування. Модуль керування зберігає це значення у пам'ять і порівнює його з попередніми значеннями. Якщо різниця між значенням поточної середньої потужності і декількома попередніми значеннями не дуже різка, він підключає волокно назад до мультиплектора чи

демультимплектора, і чекає на початок наступного тайм-слоту щоб почати перевірку наступного волокна. Якщо ж різниця велика, і втручання незаплановане оператором для цього волокна, модуль керування

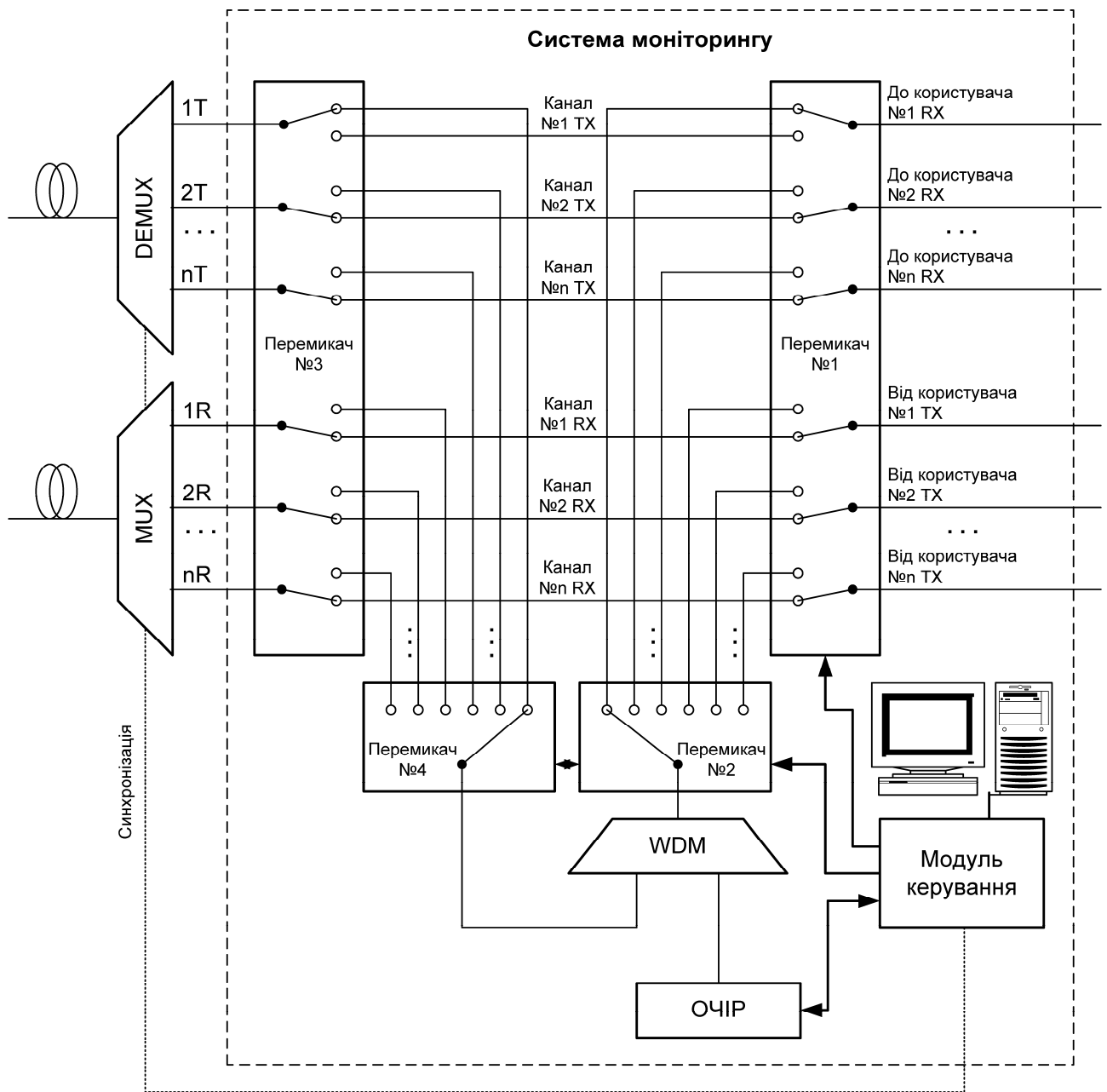


Рис. 2. Система моніторингу (другий варіант)

повідомляє комп'ютер про тривогу і помічає (у своїй пам'яті) весь цей канал як можливо пошкоджений – обидва волокна (і передаюче і приймаюче) підключаються оптичним перемикачем №1 до оптичного перемикача №2 (іншими словами – ізолюються від мережі). Це спричинить втрату пакетів, отож обидві системи (передавальна і приймальна) можуть ідентифікувати пошкодження або несанкціонований доступ. Після цього модуль керування чекає на початок наступного тайм-слоту, щоб почати перевірку наступного волокна. Але в декількох наступних перевірках помічених волокон ОЧІР буде вже передавати модулю керування повну рефлектограму. Модуль керування, усереднюючи декілька таких рефлекторів, побудує якісну рефлектограму і передасть її комп'ютеру, де програмне забезпечення може визначити і розпізнати втручання чи пошкодження. (рис. 2). Помічені волокна будуть ізольовані доки оператор не підтвердить їх підключення до мережі (рис. 3).

Послідовність тестування дуже проста. Нехай N – загальна кількість тайм-слотів. Цикл перевірки складається з двох півциклів. В першому півциклі перевіряються волокна, що йдуть від демультимплектора (будемо нумерувати їх jT , де j – номер TX тайм-слоту). Номер волокна що перевіряється F у даний момент, визначається як $C_{TX} - 1$, де C_{TX} – поточний TX тайм-слот (якщо $F=0$ то воно стає рівне N). В другому півциклі перевіряються волокна, що йдуть від мультиплектора (будемо

нумерувати їх jR , де j – номер RX тайм-слоту). Номер волокна, що перевіряється F у даний момент, визначається як $C_{RX} - 1$, де C_{RX} – поточний RX тайм-слот (якщо $F=0$ то воно стає рівне N). Якщо ми маємо N тайм-слотів, то повний цикл перевірки пройде за час $2 \cdot N \cdot t_s$, де t_s – період одного тайм-слоту.

Підключення волокна F на тестування досягається підключенням його оптичним перемикачем №1 до оптичного перемикача №2, який підключає це волокно до ОЧІР.

Другий варіант системи (складний) працює за аналогічними принципами.

За рахунок хвильового ущільнення сигналу тестування та корисних даних в одному волокні останньої милі даний варіант системи може працювати з волокнами досить великої довжини (десятки кілометрів) незалежно від тривалості тайм-слотів.

Коли волокно перевіряється, воно підключається через два оптичних перемикачі (№1, №2) до мультиплексора хвильового ущільнення, який в свою чергу підключений до ОЧІР та до двох інших оптичних перемикачів (№3, №4), які підключають його до магістралі. Кожна перевірка вирівнюється в часі з початком тайм-слоту модулем керування. Це виконується для того, щоб уникнути перекомутації волокна до ОЧІР в момент передачі даних по ньому.

При цьому загальний алгоритм роботи системи залишається такий же як і простому варіанті системи. Змінюється лише механізм підключення волокна на тестування, який в цьому випадку виконується за допомогою не двох, а чотирьох перемикачів.

Цикл перевірки, тобто принцип відбору наступного волокна для тестування в даному варіанті системи, інший. Всі волокна системи виставляються в “черзі”. Як тільки волокно, що знаходиться першим у “черзі” перестало бути активним (по ньому перестали передаватись дані), воно за допомогою перемикачів перепідключається до ОЧІР і магістралі через мультиплексор хвильового ущільнення. Після цього воно перевіряється за певний проміжок часу (в залежності від його довжини, яка заздалегідь відома системі). Після проведення перевірки, в залежності від її результатів, волокно перепідключається назад в напрямі до магістралі, або залишається ізольованим від мережі. При цьому в “черзі” це волокно переміщається у самий кінець. Першим тепер стає наступне у “черзі” волокно. Коли воно перестає бути активним, проводяться дії по його перевірці, і тепер воно переноситься у кінець “черги”. Таким чином по черзі перевіряється кожне волокно. Повний цикл перевірки пройде за час

$$T = \sum_{i=1}^n \frac{2L_i}{v}$$

де v – швидкість світла у волокні, L_i – довжина i -го волокна, n – кількість волокон.

Наприклад, в системі, в якій використовується ущільнення каналів з чотирма тайм-слотами, при використанні першого (простого) варіанта системи моніторингу, цикл перевірки буде складатися з восьми перевірок. Перші чотири перевірки будуть становити перший півцикл (рис. 4). Спочатку буде перевірятись четверте волокно, що йде до користувача – $F = 4T$ (оскільки в даний момент активний перший TX тайм-слот – $C_{TX} = 1$). Потім $1T, 2T, 3T$. Наступні чотири перевірки будуть становити другий півцикл. Першим буде перевіратись третє волокно, що йде від користувача – $F = 3R$ (оскільки в даний момент активний четвертий RX тайм-слот – $C_{RX} = 4$). Далі будуть перевірятись волокна $4R, 1R, 2R$. На цьому цикл перевірки закінчиться і все почнеться спочатку.

Висновки

Дана система достатньо гнучка оскільки може працювати з декількома оптичними волокнами. Отже вона може бути використана не тільки провайдерами, але й у офісах і підприємствах з оптичними мережами. Вона може бути використана у поєднанні з брандмауером. Також вона може бути застосована у системах де дані мають проходити через оптичні волокна, що встановлені у небезпечних місцях, оскільки дана система може точно встановити місце спроби несанкціонованого зчитування даних з волокна.

Великою перевагою даної системи над іншими є її можливість нарощування від простої до складної в залежності від росту масштабності мережі, що дозволяє зекономити кошти при використанні її в малих мережах.

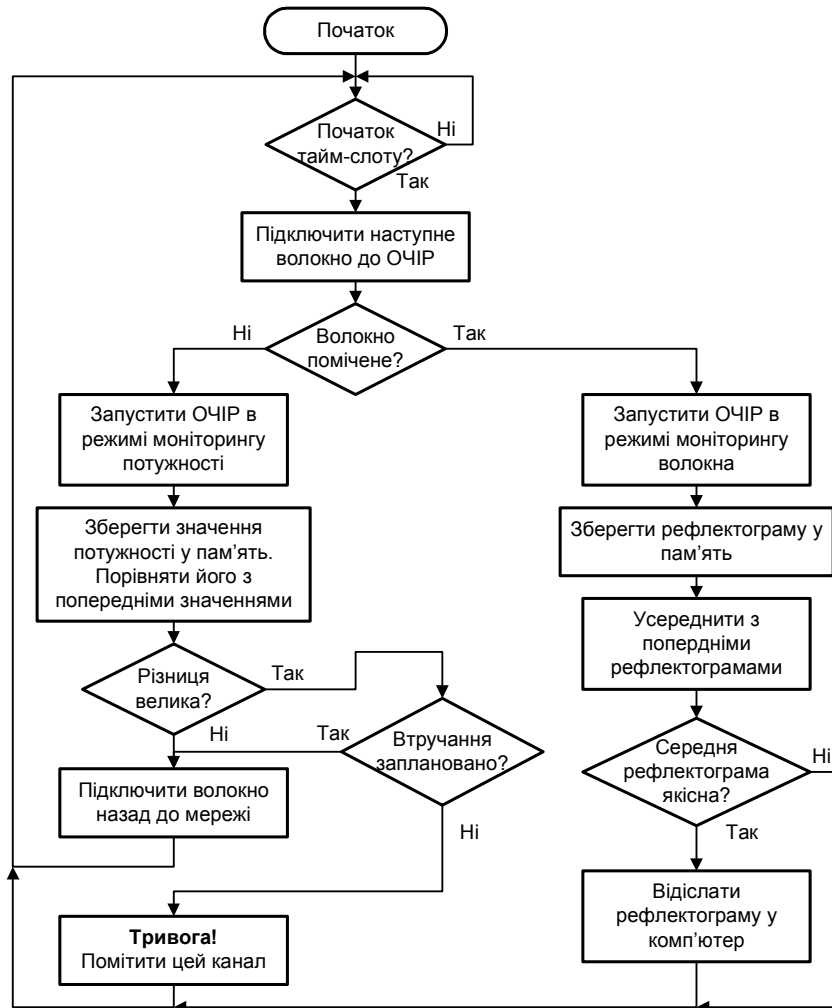


Рис. 3. Алгоритм процедури перевірки



Рис. 4. Алгоритм роботи модуля керування

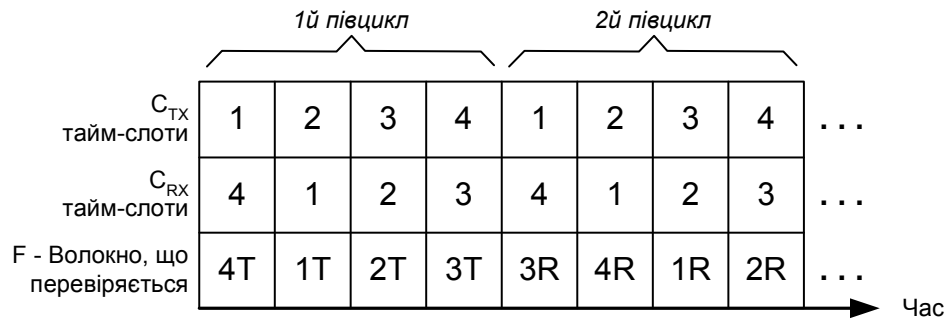


Рис. 5. Цикл перевірки (в системі з чотирма тайм-слотами)
(Т – TX волокно, R – RX волокно)

Список літератури

1. US Patent. 5,093,568 Monitoring system for fiber optic cables utilizing an OTDR for detection of signal loss and automatic location of faults in the cable. John Maycock. (March 3, 1992).
2. US Patent. 6,519,026 Optical time-domain reflectometer (OTDR). Holland, William R. (August 6, 1999).
3. US Patent. 6,141,089 Optical time domain reflectometer for measurements in optical networks with currently applied traffic signals. Thoma Peter, Hoffmann Peter. (October 31, 2000).
4. Убайдулаев Р.Р. Волоконно-оптические сети. — М.: Эко-Трендз, 2001 - 267с.
5. Иванов А.Б. Волоконная оптика: компоненты, системы передачи, измерения. — М.: Компания САЙРУС СИСТЕМС, 1999. - 671с.

Лисенко Геннадій Леонідович, заступник проректора з навчальної та науково-методичної роботи Вінницького національного технічного університету, к.т.н., доцент кафедри ЛОТ, 8(0432)447678 E-Mail: lgl@vstu.vinnica.ua

Тютюнник Ігор Ігорович, студент гр. ЛОТ-01 кафедри Лазерної та оптоелектронної техніки ІнАЕКСУ Вінницького національного технічного університету. Тел. 8(0432)355497 E-Mail: igors@vn.ua

UDC 681.3.67

C. Marinescu, N. Țăpuș

AVOIDING PITFALLS IN REAL LIFE RSA IMPLEMENTATIONS

University politehnica of Bucharest, Romania

Introduction

The RSA algorithm is used on a large scale for securing WWW traffic, e-mail, and some wireless devices. The elegance and the apparent simplicity of the RSA algorithm are due to the algebraic framework in which the algorithm was defined. But an improper use of this framework becomes dangerous, threatening the integrity and security of RSA when used either for encryption or for digital signatures. From a software developer's perspective, this is also a reason why a robust and correct implementation of the algorithm that takes all the potential pitfalls into consideration is very important.

Since RSA is based on arithmetic modulo large numbers, it can be slow in constrained environments. For example, on a heavily loaded web server, RSA decryption reduces the number of SSL requests per second that the server is capable to handle. Typically, one improves RSA's performance using special-purpose hardware, but several software implementations have also tried different improvements. In some cases, the speed improvement was invalidated by the pitfalls that were introduced. The speed of the algorithm depends mostly on the key size, the implementation type, the hardware on which the process is running, but it also depends on the protocol that has to be implemented. Many times implementers have tried to speed up the algorithm by using small public/private exponents, taking also great risks for the security of RSA into account. Using, for example, small public exponents increases the speed of the encryption and signature verification process, in comparison to the reversed processes (decryption and signing). The general algorithms that are typically used will take $O(n^2)$ time for public key operations, compared to private key operations that will need $O(n^3)$, where n is the bit number of the modulus N .

The choice of the modulus size is regarded to be one of the most important decisions in practical RSA implementations. It is quite obvious that the standard 512 bits long keys no longer provide sufficient protection, and therefore, the size should be increased. However, the complexity of modular exponentiation will grow rapidly with the size of the modulus, and thus choosing a size which combines efficient operation with long term security will become quite difficult.

The security of the algorithm is based on the assumption that the function $C = M^e \bmod N$ is hard to reverse without knowing the decryption exponent. But there is at least the theoretical possibility to find the factors of the modulus N through brute-force or by using other methods. This would expose the numbers p and q , and after using the Euclidian algorithm the decryption exponent would be revealed. The final result would be the compromised private key.

Since the publishing of the RSA, many researchers have tried to break the algorithm in order to find its weak points. Nonetheless, over the last 25 years of tests and mathematical analysis, it was possible to find different vulnerabilities, but no threat that could break the algorithm's security was found. Programming a robust version of RSA is not a trivial task as the resulted methods of attack emphasize once again the importance of a strong implementation. Trying to classify the different aims of an attacker, we can distinguish the following: 1. recovery of the private key; 2. message decryption; 3. signature forgery.

It may be possible that an attack achieving the second or third result can be mounted without the attacker actually recovering the private key. Taking the impact on the security of the algorithm into account, we can establish the following classification over the methods of attack:

- attacks that expose the private key, essentially compromising the security of the whole algorithm;
- attacks that expose just a single encrypted message or forge a signature, but leave the private key intact for further use.

Examples of methods belonging to the first category are the attack against small private exponents (in certain cases), factoring large integers. The second category includes a wide range of subtle attacks that use complicated methods to decrypt the message. We underline the fact that in these cases, there are several constraints that have to be met in order to implement a successful attack (the time has to be less than the one for the brute-force attack as specific attacks use different properties of the algorithm, imposing different restrictions). The properties, which the different methods are trying to use, can lead to a second classification:

- elementary methods of attack;

- methods against the low private exponent;
- methods against the low public exponent;
- attacks against different implementations.

Factoring Large Numbers

For some time, people believed that the security of RSA depends on the problem of factoring prime numbers. If an efficient factoring algorithm would exist, then RSA would be insecure. Recent research has proved that breaking RSA is easier than factoring prime numbers, especially when the private exponent takes small values [1]. This is also the reason why the first attack on the RSA algorithm we consider is factoring the modulus N . We refer to factoring the modulus as the brute-force attack on RSA. Although factoring RSA has improved, the current state of the art is still far from posing a threat to the security of the algorithm (the hardware improvements are not enough), especially when RSA is used properly.

The most well-known factoring algorithms can be classified into two categories: algorithms whose running time depends only on the size of the factored number, and algorithms whose running time depends on the size of the factors. The older factoring algorithms searched for the smallest factors, and were thus of the second type. However, modern algorithms tend to use other approaches, which lead to different time results. The fastest factoring algorithm of the second type is the elliptic curve method. The running time is quite fast, but its basic operations are very slow, so it is unlikely that this algorithm will be able to find 1024 bit long RSA keys in the next few years.

Algorithms of the first type are much faster, since they can use a wider range of mathematical techniques. The fastest factoring algorithm of this type is, at the moment, the General Number Field Sieve, which is running in $\exp((c + O(1)) * n^{1/3} * \log^{2/3} n)$, where $c < 2$. It is believed that this algorithm will be capable of factoring 1024 bit numbers in the next few years. Since the appearance of RSA, all the record breaking factorizations of RSA keys were based on algorithms of this type, and it is reasonable to believe that this trend will continue in the future. The best (and only) way to protect against this attack is to increase the size of the key [2].

Another factoring attack is that of Desmedt and Odlyzko, which applies equally on encryption and signing. This appears to be more practical with forging signatures rather than message decryption, since it is easier to demand the signature on different text messages instead of requesting lots of decryptions. This attack is particularly effective when the signed messages are small. The attacker is trying to factor this particular message into small primes (but the success is not guaranteed) as the signature on the message equals the product of the factors' signatures. The probability of success depends on the size of the message, not on the size of the modulus N . This is why messages should be larger than the RSA modulus and not a multiple of known values.

Elementary Methods of Attack

There are several older methods of attack, which demonstrate that an improper use of the algorithm can compromise the idea of security. The easiest method is trying to guess the message. Given the encrypted message and the public key, the attacker will guess the message and encrypt the text in order to compare the two encrypted parts. The easiest way to defend against this pseudo attack is to include a random sequence of bits at the end of the message.

Another attack method is concentrating on common used modulus. In order to avoid generating big prime numbers, implementers have tried to use common modulus for whole groups of users. A central authority was distributing the pair (e_i, d_i) to every user of the system. On a closer look at this system, one may notice that every user can use his pair of exponents to factorize the modulus. After doing this, it is possible to restore any private key based on the modulus N . The only method to avoid this scheme of attack is to deny the common use of the same modulus N for groups of users.

Another attack against RSA is called super-encryption. Simmons and Norris [3] observed that after a number of repeated encryptions the original message could eventually be recovered. This would lead to a tremendous weakness of the RSA if the number of required encryptions would be small. This is not the case if the primes are large and chosen at random, so the best method to protect the system against the super-encryption attack is to choose the prime numbers properly.

Methods Against the Low Private Exponent

This attack is a concern if the private exponent d is deliberately chosen to be small, in an attempt to improve the efficiency of decryption or signing. In such cases, the RSA is approximately 10 times faster than in the other cases. M. Wiener has demonstrated that in this case the RSA algorithm is particularly weak because the attacker has a good chance to determine the private exponent and to break the system. He has created the following theorem: *Let $N = p * q$ with $q < p < 2 * q$ and $d < \frac{1}{3} * N^{1/4}$. Given the equation: $e * d = 1 \pmod{(p - 1)(q - 1)}$, there is the possibility to efficiently recover the number d .*

The demonstration of the theorem is based on approximations of continued fractions. For a 1024 bits modulus, Wiener has proved that the private exponent has to be at least 256 bits long to prevent this attack (this might be a problem for particular implementations of the algorithm for smartcards). He has also presented other methods of preventing this attack [4]. Boneh and Durfee have proved that for a private exponent $d < N^{0.292}$, the attack cannot be prevented [5]. The best way to block this type of attack is to choose a high private exponent, even if this will increase the working time of the algorithm.

Methods Against the Low Public Exponent

In real life RSA implementations, software developers are tempted to use a small public exponent in order to speed up the encryption and/or the verification of the signature. In this case, the danger of compromising the private key is smaller. But for typical values of the public exponent, there are still several possibilities of attack.

Coppersmith published the following theorem: *Given N an integer and $f \in \mathbb{Z}_N[x]$ a monic polynomial of degree d , let $X = N^{\frac{1}{d-\varepsilon}}$, where $\varepsilon > 0$. If the pair (N, f) is known, there is the possibility to efficiently find out all numbers $|x_0| < X$ which satisfy the equation $f(x_0) = 0 \pmod{N}$.*

This offers an efficient way of calculating the roots for the $f \pmod{N}$ equation (based on a lemma by Howgrave-Graham [6]). A first application of the theorem is the improvement of an older method called the "broadcast attack". The message that had to be sent to k participants was encrypted k times with the corresponding public keys (N_i, e_i) . If, for simplicity reasons, all the members of the group would use $e_i = 3$ and if the group would consist of 3 users ($k=3$), we would obtain the following equation system:

$$\begin{cases} C_1 = M^3 \pmod{N_1} \\ C_2 = M^3 \pmod{N_2} \\ C_3 = M^3 \pmod{N_3} \end{cases}$$

Applying the Chinese Remainder Theorem (CRT) on this system, it is possible to recover the message M computing the real cube root. The attack is possible when a small public exponent is used so the best method to prevent this attack is to use bigger exponent values, or to complete the message with a sequence at the end. Hastad mentions another attack scheme directed against related messages. Completing the messages with a linear polynomial $i * 2^m$ will also not prevent the attack [7]. The use of random padding will destroy any known relation between messages, and will avoid this attack. Related messages should not be encrypted with the same RSA key.

The RSA encryption is yet not effective on small messages, especially when the public exponent is low. In particular, if $C = M^e < N$, the message can be recovered from the cipher text by ordinary root extraction. There are two options, either the public exponent should have a big value or the messages should be large enough. The second suggestion should be considered since a small public exponent is often preferred. However, the developer has to ensure that the large message he has to encrypt is not the multiple of a known value, such as a large power of 2 (this would be the case if the message would be padded at the end with 0) [8].

Another interesting attack that can be used to decrypt related messages is known also as the Franklin-Reiter method: *Assuming that the messages M_1 and M_2 satisfy the following equation $M_1 = f(M_2) \pmod{N}$, and that C_1, C_2 and the function $f \in \mathbb{Z}_N[x]$ are known, it is possible to find out the messages M_1 and M_2 for any small public exponent.*

The theorem was used also by Coppersmith to design another extended attack directed against random padded messages. This method is applicable only when $e=3$ and the padding sequence is shorter than $1/9^{\text{th}}$ of the message length. For public exponent values other than 3, the method does not work.

The partial exposure of the private key is one of the most dangerous attacks, because it leads to the recovery of the private key. Boneh, Durfee and Frankel have proved that it is possible to successfully recover the key if $e < \sqrt{N}$ and the least $n/4$ bits of d are known. This result is underlining once again just how important it is to keep the private key safe [9].

Attacks against Different Implementations

Another category of attacks, even though it is not significant as a threat to the RSA itself, is very interesting because of the pitfalls it reveals. All details such as design mistakes and small implementation bugs can bring even a good algorithm to an undesired breakdown. The famous timing attacks belong to this category. Kocher has proved that by knowing the time that is necessary to compute an encryption, it is possible to find out, bit by bit the private key [10]. To prevent this method, it is necessary to introduce random timeouts in the algorithm. Another possibility to perform this attack is to watch the power consumption (especially for

smartcards). Other successful attacks that have used other implementation specific vulnerabilities are the random faults attack [11], or the Bleichenbacher method (PKCS1) [12], etc.

Another common concern is how RSA security is affected if the pseudo primes used are, in fact, no primes at all. As a result, the factors of the modulus would be smaller, and it would be much easier for the attacker to factor it. Even if this seems to be more or less a theoretical concern, taking into account the modern methods of prime number generation, we still encourage you to pay attention to the algorithm that you are using.

The Bellare and Rogaway scheme, also called Optimal Asymmetric Encryption Padding, provides a supplement of protection to the encryption process [13]. Depending on the security level, the amount of padding is 16 bytes or more. Hence, one of the benefits is the fact that the achieved security can be related to the pseudorandom generator and the quality of the hash function.

Conclusions

It is clear that RSA is the most deployed public key algorithm today. One can find it in different operating systems (Windows, Apple, SunOS, Novell), in different hardware components (telephone systems, smartcards), or in communication protocols (S/MIME, SSL, IPSec, PKCS, S/WAN).

There are many methods of attack directed against the algorithm and many vulnerabilities that have to be considered in order to be able to implement a robust implementation of RSA. A deep analysis and a detailed knowledge of these can lead to a secure version. Some of these methods have both a theoretical and a practical value, depending on the impact that they have on the algorithm's security. Many of the presented attacks could become serious security problems if they would be mounted on raw forms of the RSA. We have also seen that there are efficient countermeasures for the real life implementations [14].

We note that in practical RSA implementations there are also other attack methods or potential threats, beyond those covered here, that should be considered from a security perspective. For example, the random key generator and the storage of the private key are vital issues in a secure implementation of RSA. There can be no secure RSA implementation without considering all the system's factors.

References

1. D. Boneh, R. Venkatesan, "Breaking RSA may not be equivalent to factoring, Advances in Cryptology" - Eurocrypt '98, Springer-Verlag, 1998, p. 59-71.
2. A. Shamir, "RSA for paranoids" - CryptoBytes, RSA Laboratories, 1995, vol. 1, no. 3.
3. G.J. Simmons, M.J. Norris, "Preliminary comments on the MIT public-key cryptosystem", Cryptologia, 1977, Vol. 1(4), p. 406-414.
4. M. Wiener, "Cryptanalysis of short RSA secret exponents", IEEE Transactions on Information Theory, 1990, vol.36, p.553-558.
5. D. Boneh, G. Durfee, "New results on cryptanalysis of low private exponent RSA", Preprint, 1998.
6. N. Howgrave-Graham, "Finding small roots of univariate modular equations revisited", Cryptography and Coding, Lecture Notes in Computer Science, Springer-Verlag, vol. 1355, p. 131-142.
7. J. Hastad, "Solving simultaneous modular equations of low degree", SIAM Journal of Computing, 1988, p.336-341.
8. B. Kaliski, M. Robshaw, "The Secure Use of RSA", RSA Laboratories, 1995, vol. 1, no. 3.
9. D. Boneh, G. Durfee, Y. Frankel, "An attack on RSA given a fraction of the private key bits", AsiaCrypt '98, Lecture Notes in Computer Science, Springer-Verlag, 1998, vol. 1514, p. 25-34.
10. P. Kocher, "Timing attacks on implementations of Diffie-Hellman, RSA, DSS, and other systems", Crypto '96, Lecture Notes in Computer Science, Springer-Verlag, 1996, vol. 1109, p. 104-113.
11. D. Boneh, R. DeMillo, R. Lipton, "On the importance of checking cryptographic protocols for faults", Eurocrypt '97, Lecture Notes in Computer Science, Springer-Verlag, 1997, vol. 1233, p.37-51.
12. D. Bleichenbacher, "Chosen cipher-text attacks against protocols based on the RSA encryption standard PKCS #1", Crypto '98, Lecture Notes in Computer Science, Springer-Verlag, 1998, vol. 1462.
13. M. Bellare, P. Rogaway, "Optimal asymmetric encryption" Advances in Cryptology, Eurocrypt '94, Springer-Verlag, 1995, p. 92-111.
14. D. Boneh., "Twenty years of attacks on the RSA cryptosystem", American Mathematical Society, 1999, vol. 46, no. 2, p. 203-213.

Cristian Marinescu, University Politehnica of Bucharest, Splaiul Independentei, No. 313, Sector 6, Bucharest, Romania, Tel.: +40-21-4100325, E-mail: ntapus@cs.pub.ro

Nicolae Țăpuș, University Politehnica of Bucharest, Splaiul Independentei, No. 313, Sector 6, Bucharest, Romania, Tel.: +40-21-4100325, E-Mail: cristian.marinescu@omicron.at

УДК 621.396.2

Г. Л. Лисенко, Р. В. Просоловський

СИСТЕМА КОНТРОЛЮ ОПТИЧНОГО КАБЕЛЮ

Вінницький національний технічний університет, Вінниця

Вступ

З розвитком телекомунікацій проблема захисту оптичного кабелю стає дедалі актуальнішою. Найбільшу увагу контролю якості роботи мережі приділяють найвпливовіші галузі, такі як банківська система (потоки інформації про трансакції коштів і т.і.), мілітарні системи (контроль якості зв'язку зі стратегічними об'єктами, система керування, тощо), магістральні мережі та інше.

Сучасний стан систем контролю оптичного кабелю

Актуальним на сьогодні є необхідність автоматизованого дистанційного моніторингу та виявлення несправностей у волоконно-оптичних лініях зв'язку (ВОЛЗ). Сама методологія передачі бітів інформації по оптичному волокну, яка полягає у розповсюдженні світлових сигналів вздовж волоконно-оптичного тракту від передавача (здебільшого напівпровідниковий лазер або світлодіод) до приймача (це, як правило, лавинний фотодіод) інформації, дозволяє як проводити контроль оптичної лінії за допомогою тестуючих імпульсів на наявність пошкоджень в ній, так і визначати віддаленість зони, в якій знаходиться причина порушення цілісності оптичного кабелю, а, відтак, і інформації, що передається.

Найчастіше у якості діагностичного та вимірювального приладу використовують оптичний часово-імпульсний рефлектометр — ОЧІР (Optical Time-Domain Reflectometer — OTDR) [1]. Робота пристроїв такого типу полягає у тому, що у волоконно-оптичний тракт запускається тестуючий імпульс, який з мірою поширення у волокні зазнає ряд відбивань, заломлень, розсіювань, загасань і т.і., що викликано неоднорідностями, втратою оптичної потужності на елементах оптичної мережі, на згинах, пошкодженнях, розтягах волокна; релеєвським загасанням, френелівським відбиттям тощо. Частина потужності з тестуючого імпульсу відбивається у зворотному напрямку, тобто назад до установки тестування ОЧІР. Далі, наприклад, за технологією Time-Of-Flight (TOF) [2], значення інтенсивності відбитого випромінювання приймається і аналізується, з подальшим записом в залежності від дискретного значення часу надходження t , що пройшов з моменту запуску контрольного імпульсу у тестоване волокно. Це значення часу є величиною прямопропорційною відстані L , на якій викликано утворення відбитого імпульсу:

$$L = \frac{c \cdot t}{2 \cdot n}, \quad (1)$$

де c — швидкість світла у вакуумі, n — показник заломлення оптичного волокна, і є величиною сталою, яка визначається виробником волоконно-оптичного кабелю. Двійка у знаменнику свідчить про те, що випромінювання пройшло подвійну відстань L (від ОЧІР до завад, та від завад до ОЧІР).

Недоліками цієї системи є недостатня швидкодія (невелика частота імпульсів, які посиляються для контролю оптичного кабелю). Це пояснюється виникненням великого рівня шумів, які заважають отримувати якісну рефлектограму. Другим недоліком є необхідність проведення тестування “темного” волокна (відключеного від передачі інформації по ньому), що є недопустимим у випадку, коли проводиться постійне контрольне тестування магістральних мереж.

Інший варіант, який вирішує проблему діагностики та тестування „темного” волокна, передбачає використання спеціальних блоків контролю потужності та генерації самостимульованого сигналу (БКП та ГСС) [3], які встановлюються по всій лінії волоконно-оптичної мережі. Кожен із таких блоків БКП та ГСС являє собою автоматизовану установку, дещо аналогічну до установки оптичного рефлектометра, але має відмінність у тому, що у тестовану оптичну лінію посиляється не просто сигнал, а певний код. Кожен з блоків БКП та ГСС отримує власний код сигналу та проводить тестування на діапазоні волокна, що підключений до його входів та виходів. Блоки БКП та ГСС з'єднуються у єдину віддалену систему контролю, яка проводить постійний автоматичний моніторинг оптичної мережі.

Не дивлячись на те, що вищеописаним пристроєм вирішується проблема “темного” кабелю, його

недоліками залишаються недостатня швидкодія, що пояснюється великим часом обирання коду та його аналізу кожним із блоків БКП та ГСС; та необхідність використання додаткової мережі, через яку ці блоки з'єднуються з віддаленою системою контролю, що в свою чергу викликає додаткові затрати, додаткове зниження швидкодії та, знову ж таки, можливість несанкціонованого доступу, наприклад, вже до цієї мережі. Таким чином, необхідно вирішити такі проблеми проведення тестування лінії за допомогою ОЧП:

- підвищення швидкодії тестування оптоволоконної лінії, яка залежить від багатьох факторів: виникнення різного роду шумів та завад (релеївські розсіювання, френелівські відбивання тощо), та параметрів електроніки самого приладу ОЧП (це можуть бути “мертві” (ще називають “сліпі”) зони рефлектометра, які визначаються “засвіченістю” фоточутливого елемента; швидкістю аналого-цифрового перетворення; швидкістю запису у пам'ять (буферизація), швидкістю обміну інформацією, що передається, та інше);
- підвищення рівня захисту контрольованого оптоволоконного тракту, що пов'язано як з швидкістю проведення тестування лінії, так і з методами та алгоритмами його проведення.

Вдосконалена система ОЧП

Для вирішення поставлених задач пропонується вдосконалений ОЧП. Система складається з блоку керування (БК) 1 (рис. 1), який з'єднаний з блоком ОЧП 2. Оптичні вихід та вхід блоку ОЧП 2 під'єднані відповідно до прямого входу та зворотного виходів оптического направленного відгалужувача 7. Прямий вихід відгалужувача 7 разом із основними (корисними) волокнами 8 потрапляє на один із входів оптического мультиплексора 9. Оптичний мультиплексор/демультиплексор (MUX/DEMUX) 9 через контрольоване одномодове оптичне волокно 10 з'єднується з оптичним мультиплексором/демультиплексором (MUX/DEMUX) 11, виходи якого відповідно під'єднуються до основних (корисних) волокон 12 та атенюатора тестуючої довжини хвилі випромінювання 13.

Блок ОЧП 2 може бути або спеціально підбраною базовою моделлю (наприклад, універсальна система тестування EXFO “FTB-300” [4], Syrus System “CMA5000-OTDR” [5] тощо), або дещо спрощеним з метою економії, окремо сконструйованим блоком (рис. 2). Такий блок ОЧП 2 має імпульсний лазерний діод (ЛД) 5, однаковий вихід якого подається до направленного відгалужувача 7; однаковий вхід від направленного відгалужувача на фотоприймач у виконанні лавинного фотодіода (ЛФ) 6. ЛФ 6 з'єднується зі швидким аналого-цифровим перетворювачем (АЦП) 4, вихід якого з'єднується з оперативним запам'ятовувальним пристроєм (ОЗП) 3. Блок ОЧП 2 має також двонаправлений електричний вхід, через який з'єднується з БК 1.

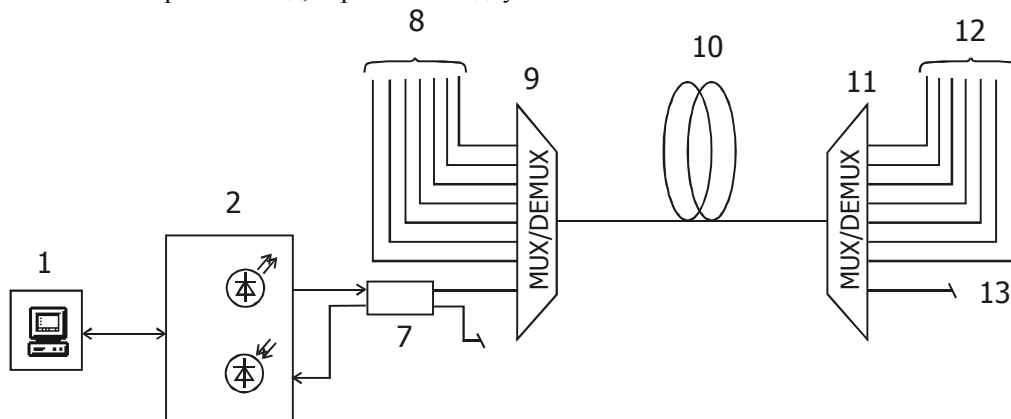


Рис. 1. Структурна схема системи контролю оптичного кабелю

Принцип роботи системи контролю оптичного кабелю

Система працює наступним чином (рис. 3). БК 1, маючи сигнал готовності і роботи лінії, подає в оптичний рефлектометр ОЧП 2 через шину керування (ШК) на вхід ЛД 5 імпульс тривалістю τ , а також одночасно починає відлік часу t . У відповідності до цього імпульсу ЛД 5 випромінює когерентний пучок променів тривалістю τ на прямий вхід направленного оптического відгалужувача 7. Відгалужувач 7 направляє випромінювання з прямого виходу на вхід MUX/DEMUX 9. Відповідно до використовуваної технології ущільнення (WDM або DWDM) [6], випромінювання разом із корисними сигналами 8

мультиплексується і направляється через вихід MUX/DEMUX 9 у тестоване одномодове оптичне волокно 10. З розповсюдженням сигналів (корисних та тестуючого) по волоконно-оптичному тракті, випромінювання загасає, заломлюється, відбивається від неоднорідностей (за законом Френеля) тощо, в наслідок чого виникають зворотні розсіювання як корисних так і тестуючого сигналу. Далі ці сигнали потрапляють на вхід MUX/DEMUX 11, де розділяються на корисні вихідні канали та канал тестування. Тестуюче випромінювання загасає на оптичному атенуаторі 13.

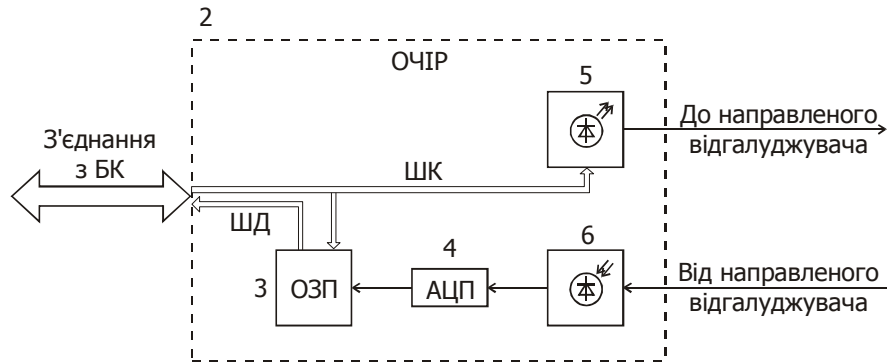


Рис. 2. Конструкція ОЧІР 2

Та частина випромінювання, що відбилася на неоднорідностях оптичного волокна 10 повертається на вхід MUX/DEMUX 9 і відбирається на зворотний вхід направленої відгалужувача 7. Далі випромінювання заломлюється на однонаправлений вхід блоку ОЧІР 2 і потрапляє в ЛФ 6, викликаючи виникнення струму, аналогове значення якого перетворюється швидким АЦП 4 в цифрове і записується в ОЗП 3 в залежності від часу t , який пройшов від початку проведення тестування. Аналогічних відбивань у лінії 10 виникає безліч, і вони продовжують проходити вищеописаний шлях і, таким чином, потрапляти у вигляді цифрового значення струму на ЛФ 6 в ОЗП 3 в залежності від часу приходу t . Таким чином, формується таблиця значень інтенсивності відбитого випромінювання (перша колонка) і часу їх надходження (друга колонка). Ця таблиця і є інтерпретацією рефлектограми, де перша колонка є рівнем загасання у волокні α , дБ, а друга — віддаленість цього загасання, яка визначається за формулою 1. Чим більша кількість рядків утвореної у ОЗП 3 матриці даних, тим якісніше вимірювання за рахунок більшої роздільної здатності рефлектограми. Значення кількості рядків матриці даних залежить від функціональних можливостей пристроїв, що використовуються в схемі (електроніка, параметри ЛФ 6 тощо).

Аналіз отриманої рефлектограми виконується програмно в БК 1 шляхом порівняння отриманих даних з допустимими значеннями. У разі, якщо рефлектограма задовольняє діапазон значень, ОЗП 3 рефлектометра очищується від даних. Очікується час $t \geq T$ (час затримки між імпульсами), далі значення t обнуляється і знову починається відлік часу t з одночасно посланим сигналом на ЛД 5. Знову відбувається вищеописаний цикл контролю до того моменту, поки значення рефлектограми буде незадовольняти діапазон допустимих значень. Тоді БК 1 за необхідності дає сигнал про припинення передачі корисних даних 8 (у випадку, коли важливо зберегти конфіденційність інформації), визначає віддаленість місця пошкодження волокна L , визначає характер пошкодження, оповіщає персонал про поломку та її віддаленість та проводить інші необхідні функції.

Будова даної системи враховує використання направленої оптичного відгалужувача 7. Такий відгалужувач повинен бути спеціальним, оскільки є досить важливим у конструкції для того, щоб не викликати "засвічення" ЛФ 6 відбитим від початку волокна тестуючим променем (стрибок 14 на рефлектограмі, рис. 4, а). Такі "засвічення" ЛФ 6 вимагають часу, для того, щоб інерційний ЛФ 6 був здатний продовжувати тестування. Очевидно, що у випадку тестування лінії з затримкою між тестуючими імпульсами $T=1$ мкс з їх тривалістю $\tau=10-100$ нс ЛФ 6 не буде чутливим до картини рефлектограми. З іншого боку, відгалужувач 7 повинен повністю подавати зворотне випромінювання (те, що повернулось з тестованої лінії) на ЛФ 6, без його потрапляння на однонаправлений вихід ЛД 5, щоб не викликати появу самостимульованого випромінювання ЛД 5, яке у даному випадку буде шкідливим. Використання направленої оптичного відгалужувача, запропонованого у [7] або інших відгалужувачів, які використовують поляризацію випромінювання дозволить уникнути даної проблеми.

Оптичний атенуатор 13, поставлений в умовному кінці тестованої лінії дозволяє проводити затухання оптичного випромінювання на довжині хвилі, що використовується для тестування лінії (згідно з WDM- або DWDM-технологією ущільнення). Таке затухання необхідне для того, щоб

уникнути піку в рефлектограмі 16 на кінці досліджуваної лінії, що додатково дозволить зменшити кількість помилкових (шкідливих) сигналів на отриманій рефлектограмі (рис. 4, а, б).

Детальний програмний аналіз рефлектограми (рис. 3) полягає у порівнянні отриманих значень затухання 15 в тестованому волокні з діапазоном допустимих відхилень значень затухання 18.

Модель роботи системи контролю оптичного кабелю

Швидкості реакції ОЧІР, що представлений у [1] в першу чергу не достатньо за рахунок того, що сама технологія не передбачає швидкість тестування більшу за мінімальну віддаленість подвійної відстані до кінця лінії. Значення подвійної відстані до кінця лінії встановлюється формулою (1), і з довжиною ділянки контрольованого оптичного кабелю в 120 км становить значення від 4 мс, що є достатньо великим значенням для проведення надійного та швидкого контролю оптичного кабелю.

Запропонована система контролю оптичного кабелю дозволяє збільшити частоту тестування оптоволоконної лінії до 1 МГц, або до інтервалу часу в $T=1$ мкс з тривалістю імпульсу в 1 нс, за рахунок того, що пріоритетним є посилення тестуючих імпульсів у волоконно-оптичний тракт, а не детальний аналіз картини рефлектограми (рис. 5), такий аналіз доцільно робити вже після виявлення сигналу про виявлення пошкодження або підозри про пошкодження волоконно-оптичного тракту.

На рис. 5а умовно зображено оптичне волокно, яке складається з двох ділянок рівних його довжині L . Ці дві ділянки відповідають двом напрямкам розповсюдження оптичного випромінювання у волокні (в прямому та зворотному напрямку). Припустимо, що перша ділянка 20-23 має незначне пошкодження 21, та пошкодження 22, яке викликане несанкціонованим доступом. Аналогічно, при зворотному розповсюдженні випромінювання, яке відбилося від умовного кінця лінії на відстані L , на ділянці волокна 24-27 знаходяться відповідно пошкодження 25 та 26.

На рис. 5б зображено графік залежності оптичних імпульсів, що подаються ЛД 5 у оптичне волокно. Затримка між імпульсами складає T .

На рис. 5, в зображено графік відображених імпульсів, що надходять на ЛФ 6 впродовж тестування лінії системою контролю оптичного кабелю (рефлектограма з тестованого оптичного волокна).

У випадку, якщо б використовувалась установка ОЧІР [1], то сигнал тривоги з'явився б із затримкою в $2 \cdot T_1$ після посилення імпульсу, що може бути надзвичайно великою величиною. Натомість, запропонована система контролю оптичного кабелю виявить подію пошкодження кабелю з максимальною затримкою в тривалість затримки між імпульсами T , яка на декілька порядків нижча за вищенаведене значення. Для даної системи є пріоритетним аналіз рефлектограми на наявність перевищення її значень в допустимому діапазоні між рівнем шумів (α_n) та статистично встановленим максимальним значенням амплітуди сигналу (α_{max}), що повертається з тестованої лінії. Виявлення події пошкодження з точністю T можливе лише при безперервному послідовному тестуванні лінії.

Детальний аналіз типу пошкодження та його віддаленості, подібно до технології установки ОЧІР [1] можна проводити в режимі тривоги, що описано в блок-схемі на рис. 3, з попереднім відключенням лінії.

Режими роботи системи контролю оптичного кабелю

Як вже відзначалось раніше, наведена система контролю оптичного кабелю має два режими роботи:

1. Режим неперервного послідовного тестування. Це вважається основним режимом роботи системи контролю оптичного кабелю. В даному режимі виконується перша частина блок-схеми роботи, що наведена на рис. 3. Система з інтервалом в T надсилає тестуюче оптичне випромінювання тривалістю τ . В цей же момент система проводить попередній швидкий аналіз рефлектограми на предмет її збереження у допустимих межах. У випадку, якщо рефлектограма не задовольняє встановлений діапазон, система переходить у другий режим роботи.
2. Режим тривоги і детального аналізу рефлектограми. У цьому режимі система працює у випадку, якщо картина рефлектограми не задовольняє допустимі межі. БК 1 системи забороняє подальшу передачу корисних даних 8 по волокну та проводить аналіз пошкодження з виявленням його типу та його віддаленості.

Підготовка роботи системи контролю оптичного кабелю

Для вищепов'язаної повноцінної роботи системи контролю оптичного кабелю важливу роль відіграє підготовка її роботи, тобто вірно визначений діапазон допустимих значень рефлектограми (рис.6).

Для визначення діапазону значень рефлектограми, отриманої з досліджуваного оптичного волокна рис. 6, а, необхідно при встановленні запропонованої системи контролю оптичного кабелю провести її роботу у другому режимі, тобто, у режимі детального аналізу даних. У цьому випадку вважатиметься, що

контрольоване волокно є готовим до проведення контролювання і на ньому відсутні будь-які пошкодження або несанкціоновані доступи, які варті уваги системи. Далі система проводить послідовний збір даних у вигляді послідовності детальних рефлектограм рис. 6, б, які матимуть певні стрибки в умовних точках 30, 31, 32 та 33, які, наприклад, є відповідно початком тестованої лінії, двома її незначними пошкодженнями (згинами, або іншими присутніми на лінії пристроями, які вносять певні відображення світлового сигналу) та кінець лінії. Рефлектограми збираються та аналізуються БК 1 системи контролю оптичного кабелю. БК 1 на основі отриманих даних створює картину діапазону допустимих значень 34 рефлектограми (рис 5, в). Далі БК 1 проводить множення діапазону заданих значень 34 рефлектограми у відповідності до обраної частоти проведення тестування лінії Т і таким чином формує кінцевий діапазон допустимих значень 36.

Система з часом може виявляти „фальшиву” тривогу, що може пояснюватись розширенням оптичної мережі, зміни характеристик волокна з часом тощо, але цьому можливо запобігти, якщо проводити перевизначення діапазону значень рефлектограми з часом, або після розширення оптичної мережі.

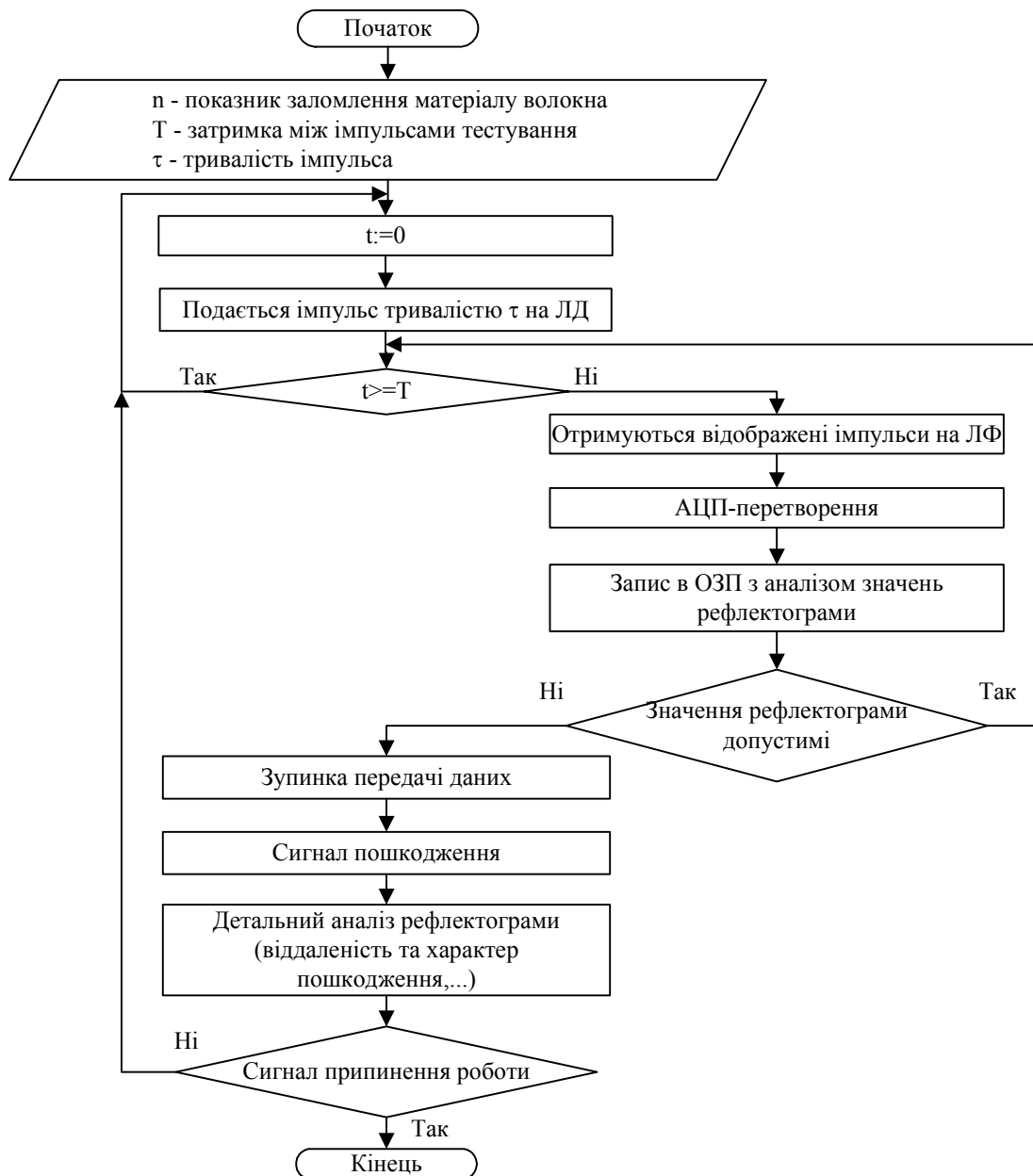


Рис. 3. Блок-схема запропонованого алгоритму тестування оптичного кабелю

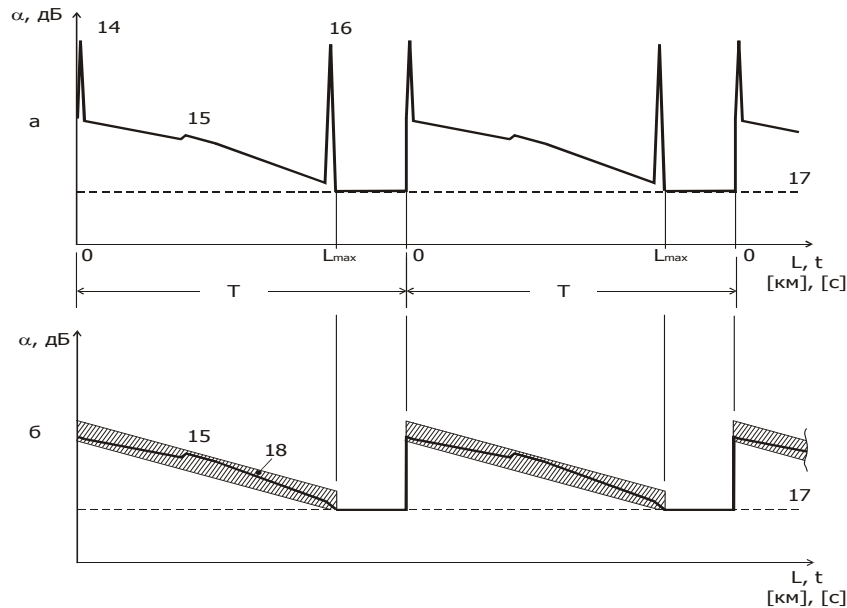


Рис. 4. Приклади типових рефлектограм, отриманих при тестуванні оптичного волокна
 а – типова рефлектограма відомого прототипу, б – типова рефлектограма, отримана при проведенні тестування досліджуваної (контрольованої) лінії вдосконаленим ОЧР

Визначення системою контролю оптичного кабелю характеру пошкодження лінії

Важливим елементом системи контролю оптичного кабелю є характер пошкодження, адже від типу пошкодження (чи то злам, чи то згин, чи то вібрація, чи то несанкціонований доступ тощо) залежить швидкість та ефективність його усунення, що є основними параметрами в оптичних мережах з високим рівнем захищеності.

Встановлення характеру та віддаленості пошкодження контрольованого оптичного кабелю здійснюється в другому режимі роботи системи контролю оптичного кабелю (рис. 3), коли ввімкнено сигнал тривоги і проводиться послідовний детальний аналіз рефлектограм, які продовжують надходити з тестованого волокна. При цьому, БК 1 системи контролю оптичного кабелю за характером зміни кривої даних рефлектограми програмно визначає тип пошкодження та його віддаленість за формулою (1). Пошкодження можуть бути або типовими, що зберігаються у пам'яті БК 1, або статистично зібраними за аналогією до попереднього пункту по підготовці до роботи системи контролю оптичного кабелю: проводяться примусові згини, злами та втручання в контрольоване оптичне волокно з подальшим отриманням діапазону значень рефлектограми та описом типу пошкодження. Такий підхід робить систему контролю оптичного кабелю більш гнучкою в аналізі типу пошкодження яке може виникнути на лінії.

Висновок

Запропонований варіант системи контролю оптичного кабелю можна використовувати у якісних та швидких оптичних мережах обміну інформації, де першочергово важливою є конфіденційність інформації, що передається. У разі виникнення несанкціонованого доступу, БК 1 припиняє будь-яку передачу даних і оповіщає персонал про пошкодження лінії, про характер даного пошкодження та про його віддаленість. Установка проводить тестування лінії з затримкою в 1 мкс, тому у разі несанкціонованого доступу, якщо доступ у кращому випадку тривав 1 мкс, то з оптичного волокна, інформація по якому передається на швидкості 10 Гбіт/с, буде зчитано 10 кбіт корисної інформації, що становить допустимі значення.

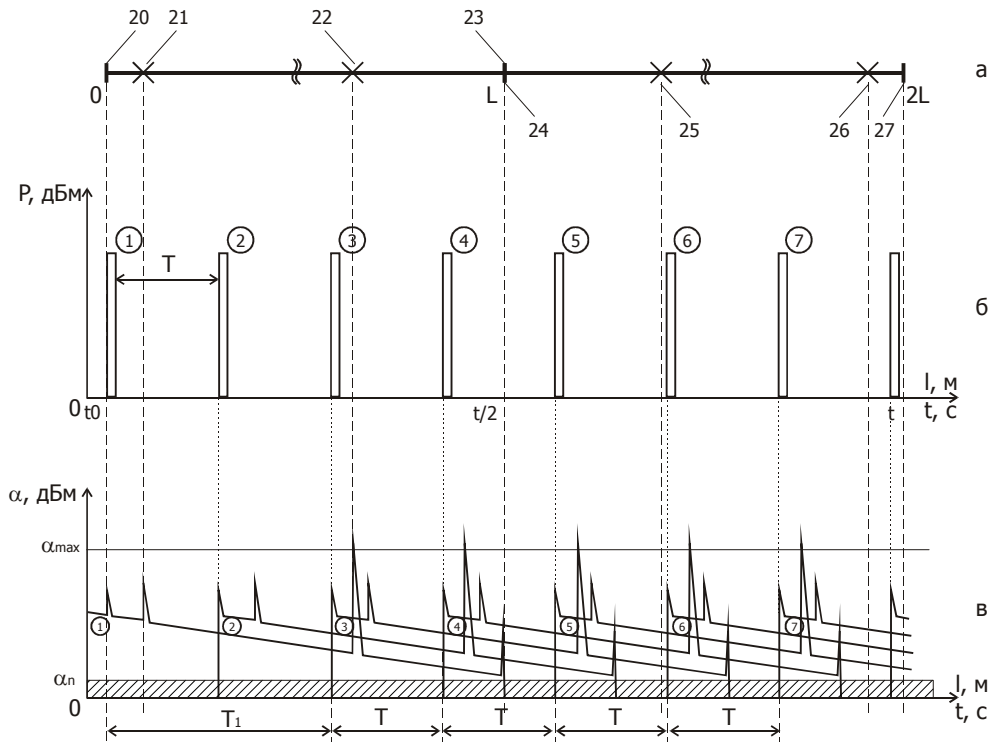


Рис. 5. Модель роботи системи контролю оптичного кабелю

а – тестоване оптичне волокно з подвійною довжиною (прямий та зворотній напрям поширення оптичного пучка);
 б – графік залежності імпульсів, що подаються ЛД 5 у оптичне волокно від часу їх посилання;
 в – графік відображених імпульсів, що надходять на ЛФ 6 впродовж тестування лінії системою контролю оптичного кабелю.

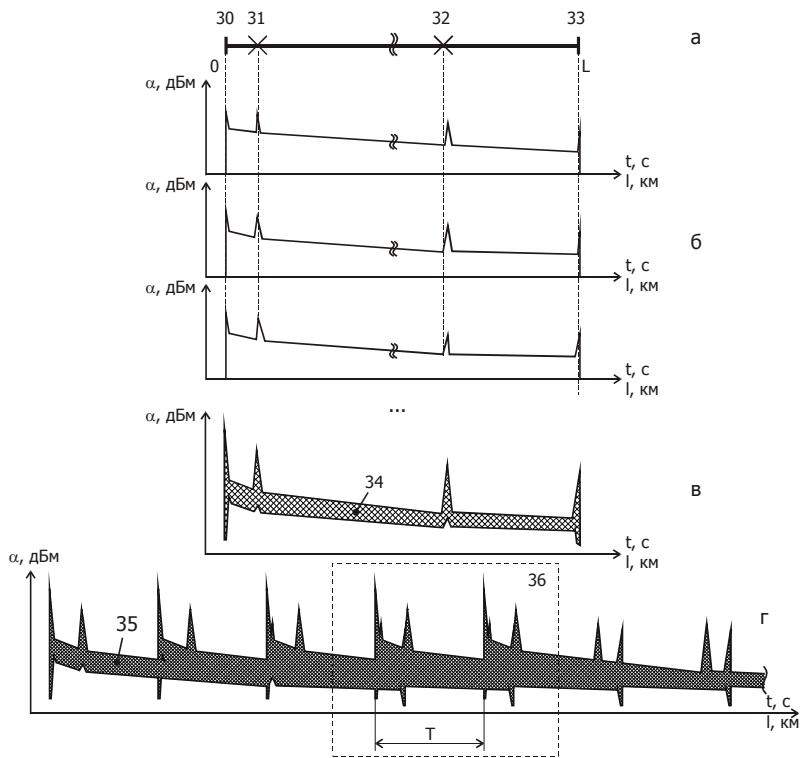


Рис. 6. Приклад визначення діапазону допустимих значень рефлектограми:

а – контрольоване оптичне волокно довжиною L ; б – рефлектограми, отримані при тестуванні контрольованого оптичного волокна; в – діапазон даних рефлектограм з урахуванням періодичності T посилання тестуючих ліній імпульсів.

Список літератури

1. Иванов А.Б. Волоконная Оптика: компоненты, системы передачи, измерения. — М.: Компания САЙРУС СИСТЕМС, 1999, 671с.
2. US Patent. 5,745,437 Method and apparatus for coherent burst ranging. Wachter , et al. (April 28, 1998).
3. US Patent. 5,859,716 Self-stimulation signal detection in an optical transmission system. O'Sullivan , et al. (January 12, 1999).
4. <http://www.adp.ru/KATALOG/EXFO/>
5. <http://www.sygus.ru/>
6. Р.Р.Убайдулаев Волоконно-оптические сети. — М.: Эко-Трендз, 2001, 267с.
7. US Patent. 4,746,184 Light coupler for optical reflectometry. Gang. (May 24, 1988).

Лисенко Геннадій Леонідович, заступник проректора з навчальної та науково-методичної роботи Вінницького національного технічного університету, к.т.н., доцент кафедри ЛОТ, 8(0432)447678
E-Mail: lgl@vstu.vinnica.ua

Просоловський Руслан Васильович, студент гр. ЛОТ-01 кафедри Лазерної та оптоелектронної техніки ІнАЕКСУ Вінницького національного технічного університету.

УДК 681.324

О. М. Хошаба, О. А. Дячок

ПРОБЛЕМИ ТА ЕФЕКТИВНІ РІШЕННЯ ПОБУДОВИ СУЧАСНИХ КОРПОРАТИВНИХ КОМП'ЮТЕРНИХ МЕРЕЖ

Вінницький національний технічний університет, Вінниця

Вступ

В наш час комп'ютерні мережі стали невід'ємним атрибутом сучасних підприємств, інструментом для успішного ведення справ в умовах високої конкуренції і насичення інформаційних потоків. Успішно застосовувати передові інформаційні технології дозволяють сучасні програмні і технічні засоби. В цьому випадку важливо зробити правильний стратегічний вибір шляху розвитку мережі свого підприємства. Тому, необхідно мати всю інформацію про сучасні мережні технології, знати їх можливості. Питання ефективного проектування корпоративних комп'ютерних мереж є дуже актуальним, оскільки вдало вибрана та спроектована топологія має більш суттєвий вплив на стабільність мережі, ніж функціонування технологій фізичного рівня.

Вирішення проблеми стабільності функціонування корпоративних комп'ютерних мереж

Проектування корпоративної мережі необхідно починати з найнижчого рівня – фізичного середовища передачі інформації. До основних функцій фізичного рівня належать обробка бітів і байтів, оптимальний вибір пропускної здатності лінії зв'язку, взаємодія з середовищем передачі інформації.

Стабільність роботи каналів зв'язку багато в чому визначається надійністю передачі потоків даних в мережі. Пошкодження фізичного рівня, що часто виникають у комп'ютерних мережах призводять до змін, які викликають необхідність оновлення таблиць маршрутизації. Порівняно з цим, топологія (структура) мережі має суттєвіший вплив на стабільність функціонування програмних засобів, ніж підтримка технологій фізичного рівня (ATM, Frame Relay або інші). Відомо, що вдало вибрана топологія є надійним показником мереж, що стабільно функціонують. Існують непоодинокі випадки, коли проблему помилково спроектованої мережі намагаються вирішити за рахунок придбання більш продуктивних маршрутизаторів, різноманітних доопрацювань адресної схеми або вдосконалених протоколів маршрутизації.

Методи побудови корпоративних комп'ютерних мереж

Велика за масштабами комп'ютерна мережа може бути поділена на декілька, порівняно невеликих частин, кожна з яких можна використовувати окремо від інших.

Також, з більшості проведених досліджень відомо, що якісно спроектована мережа являє собою ієрархічну структуру, тобто, мережу, поділену на декілька рівнів. Кожний рівень представляє собою окрему область з чітко визначеною метою.

Концепція ієрархічних мереж нагадує методологію побудови моделі протоколів OSI, згідно з якою процес взаємодії між комп'ютерами розбивається на декілька функціональних рівнів, які виконують визначене коло задач. Рівні ієрархічної моделі повинні по можливості точно відповідати поставленим перед ними задачами. Спроба делегування будь-якому визначеному рівню надто великої кількості функціональних задач як правило призводить до безладу. Це ускладнює процес створення та супроводження комп'ютерних мереж.

В більшості випадків ієрархічна модель комп'ютерної мережі передбачає визначення трьох рівнів. Кожен рівень ієрархічної мережі виконує окремі функціональні задачі:

ядро (core) мережі відповідає за високошвидкісну передачу мережного потоку даних в первинне призначення пристрою, що входить в ядро мережі. До основної функції ядра належить комутація пакетів;

на рівні розподілу (distribution layer) відбувається складання маршрутів та агрегація потоків даних;

рівень доступу (access layer) відповідає за формування мережного потоку даних, здійснює контроль точок входу в мережу та представляє інші служби граничних пристроїв.

Існують два основних принципи розробки структури мережі: розміри ділянки мережі, на які впливає зміна топології, мають бути строго обмежені; при цьому бажано, щоб дана ділянка мережі була як можна меншою. маршрутизатори (а також інші мережеві пристрої) повинні обробляти максимально можливий об'єм інформації.

Кожний з цих принципів може бути реалізований за допомогою додавання, яке здійснюється на рівні розподілу ієрархічної моделі мережі. Отже, область збіжності протоколу маршрутизації має бути обмежена рівнем розподілу.

Рівні ієрархічної моделі повинні відповідати поставленим перед ними задачам.

Одним з найсуттєвіших недоліків ієрархічної структури мережі є потенціальна можливість виникнення одиничних точок відмови фізичного рівня. Також вважається [1,2], що для вимогливішої за організацією ієрархічної структури мережі, ймовірність виникнення пошкодження одного пристрою або каналу передачі даних призведе до суттєвих порушень в її роботі.

Загальна характеристика топологічних рівнів корпоративної мережі.

Ядро

Перед ядром мережі поставлена задача комутації пакетів (switching packets) і все, що забирає обчислювальні ресурси пристроїв ядра або збільшує затримку комутації пакетів, повинно розглядатися, як загроза продуктивності всієї мережі. Саме в ядрі мережі сконцентрована вся її обчислювальна потужність. Дві основні стратегії, що дозволяють досягти максимальної продуктивності, такі:

в ядрі не повинні реалізовуватися мережні правила;

кожен пристрій ядра повинен мати можливість доступу до кожного пункту призначення мережі.

Рівень розподілу

Перед рівнем розподілу мережі поставлено три чітко сформульовані задачі: ізоляція наслідків зміни топології; керування розміром таблиці маршрутизації; агрегація мережевих потоків даних.

Для досягнення цих цілей використовуються дві основні стратегії, що реалізуються на рівні розподілу:

додавання маршрутів;

мінімізація каналів, що з'єднують рівень розподілу з ядром мережі.

Рівень доступу

Перед рівнем доступу поставлено три основні задачі: формування мережевих потоків даних; контроль доступу до мережі; виконання інших функцій для мережевих пристроїв.

Пристрої рівня доступу з'єднують високошвидкісні канали локальних мереж з каналами глобальної мережі які формують потоки даних на рівень розподілу. Пристрої рівня доступу являють собою видимі частини мережі.

Особливості ієрархічної маршрутизації

Ієрархічна маршрутизація є найефективнішим способом проектування корпоративної мережі завдяки тому, що:

ієрархія передбачає розділення однієї великої проблеми на декілька менших, кожна з яких може бути

вирішена окремо від інших;

дозволяє ефективно зменшити розмір ділянки мережі, на який впливає зміна топології;

дозволяє зменшити об'єм інформації, яку оброблює і підтримує маршрутизатор;

створює передумови для проведення додавання маршрутів та агрегації потоків даних.

Ієрархічну маршрутизацію забезпечують сучасні інформаційні технології, а саме – протоколи маршрутизації RIP, OSPF, IGRP та інші.

Для прикладу розглянемо протоколи маршрутизації, RIP та OSPF.

OSPF – відкритий протокол, що базується на алгоритмі пошуку найкоротшого шляху (Open Shortest Path First - OSPF) і є протоколом маршрутизації, розробленим для мереж IP робочою групою Internet Engineering Task Force (IETF). Цей протокол базується на алгоритмі SPF. Алгоритм SPF деколи називають алгоритмом Dijkstra за іменем автора, який його розробив. OSPF є протоколом маршрутизації з повідомленням стану про канал (link-state). Це означає, що він вимагає відправлення повідомлень про стан каналу (link-state advertisement - LSA) на всі роутери, які знаходяться в межах однієї і тієї ж ієрархічної області. У повідомленнях LSA протоколу OSPF існує інформація про відключені інтерфейси та змінні інших показників. З мірою накопичення маршрутизаторами OSPF інформації відносно стану каналу, вони використовують алгоритм SPF для розрахунку найкоротшого шляху до кожного вузла

корпоративної мережі. До таких же протоколів, які повідомляють стан каналу зв'язку за допомогою вектора відстані, належать RIP та IGRP. Тому маршрутизатори, що використовують протоколи на основі алгоритма вектору відстані, відправляють всю або частину своєї таблиці маршрутизації в повідомлення про коректування маршрутів. Однак, ця інформація передається тільки для своїх „сусідів”. На відміну від RIP, OSPF може працювати в межах деякої ієрархічної системи.

RIP відноситься до протоколів маршрутизації, який був розроблений для універсального протоколу PARC Хеґох і використовувався в комплекті протоколів XNS. Тому, кожен запис даних в таблицю маршрутизації RIP вміщує в себе різноманітну інформацію, включаючи кінцевий пункт призначення, наступну пересилку на шляху до цього пункту призначення та показчик (metric). Показчик позначає відстань до пункту призначення, виражене кількістю пересилок до нього. В таблиці маршрутизації може знаходитися також інша інформація, в тому числі різноманітні таймери, які пов'язані з даним маршрутом. RIP підтримує тільки найкращі маршрути до пункту призначення. Якщо нова інформація забезпечує кращий маршрут, то ці дані замінюють стару маршрутну інформацію. Зміни в топології мережі можуть викликати значні зміни в маршрутах, наприклад, якщо якийсь новий маршрут стає кращим до конкретного пункту призначення. Коли мають місце зміни в топології мережі, то вони відображаються в повідомленнях про коректування таблиці.

Також RIP визначає ряд характеристик, призначених для стабільнішої роботи в умовах, коли топологія мережі швидко змінюється. До них відносяться: обмеження кількості пересилок, тимчасові затримки змін (hold-downs), розщеплення горизонту (split-horizons) та коректування зміни (poison reverse updates).

Висновки

У практичній діяльності важливо реалізувати ієрархічну структуру на початку життєвого циклу мережі, тобто тоді, коли корпоративна мережа ще не досягла великих розмірів. При збільшенні розмірів корпоративної мережі все важчою буде перебудова її структури. Однак, правильне проектування корпоративної мережі допоможе уникнути значних витрат часу на ліквідацію мережевих пошкоджень в майбутньому.

Тому, сучасні інформаційні технології, які необхідно використовувати при побудові корпоративних комп'ютерних мереж представляють собою:

методи створення комп'ютерних мереж, які ґрунтуються на міжнародних стандартах провідних організацій світу;
інтелектуальні програмні засоби побудови корпоративних комп'ютерних мереж – протоколів маршрутизації.

Список літератури

1. «Принципы проектирования корпоративных IP-сетей», Cisco Press
2. «CISCO Internetworking Technology Overview»

Хошаба Олександр Мирославович, к.т.н., доцент, доцент кафедри інтелектуальних систем, доц., к.т.н., Вінницький національний технічний університет, Хмельницьке шосе, 95, Вінниця, 21021, Україна, тел.: (0432) 43-78-80, E-Mail: hoshaba@hoshaba.org

Дячок Олеся Анатоліївна, студентка гр.ІС-02, Вінницький національний технічний університет, Хмельницьке шосе, 95, Вінниця, 21021, Україна, тел.: (0432) 43-78-80, E-Mail: olesya@vstu.vinnica.ua

UDC 681.3.06

V. Hirayama, W. Jaimes Salcedo, F. J. Ramirez-Fernandez

EMBEDDED SYSTEM TO CLASSIFICATE HEAT POWER OF A FUEL GAS AND THE QUALITY OF ALCOHOL

Escola politécnica, cidade universitária, São Paulo

Introduction

Recently in academic and industrial fields there was a considerable increase in interest on electronic noses, because the possibility to make direct measures with few refinements and because its facility of implementation [1]. An electronic nose has many applications, for example: liquid and solid smell recognizers, perfumes and chemical reagents [2-4], illness detection [5], breath alcohol measurement [1], quality of potable water monitoring [6], among others.

An electronic nose in almost all cases is implemented with Artificial Neural Networks, because its noise robustness that can exist in analyzed samples [7-8], moreover it has a great capability of generalization, that lead to new samples inferences, corrects in most of cases, outside the training samples set. The application implemented in this work is the classification of heat power of a fuel gas.

When there are many variables to be analyzed, a principal component analysis is in general applied to minimize the dimensionality of the input space, keeping as most information as possible in data [9-10].

The six gas sensors measures in our apparatus were analyzed into two approaches: the use of raw data, and the use of data processed by a principal component analysis with fewer sensors.

Traditional methodologies to measure the heat power of a fuel gas can be divided in three categories [11]: combustion of a gaseous sample inside a calorimetric bomb, combustion of a gas in an open flame of a gas burner, and combustion without a flame on a catalyst. Those methods require in general expensive machinery. An embedded system of classification can be a low cost alternative in classification of the heat power of a given fuel gas.

This work is in its initial phase. The proposed objective is to develop an embedded system to recognize the heat power of a fuel gas. A matrix of six sensors will be used to measure the concentration of different gases. However, as the measure is in its initial stage, synthetic data have been generated to simulate the sensors behavior. These data was applied in some artificial neural network topologies and an fuzzy inference system, and they are being implemented in hardware, using DSP's and micro controllers.

Methodology and Experimental Results

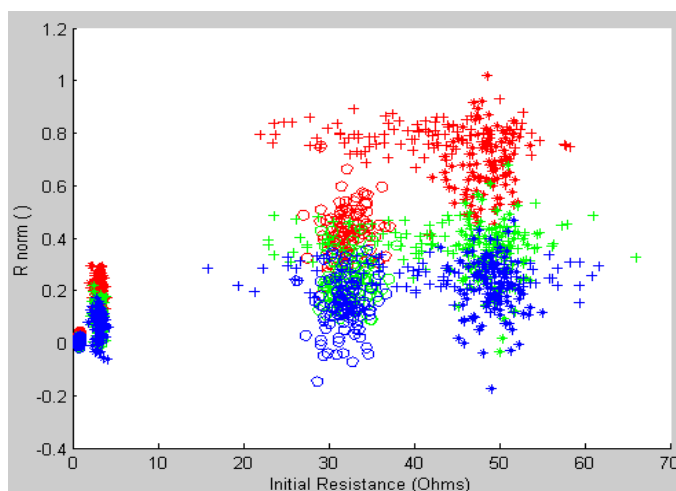


Fig. 1. Synthetic data used to create the ANN and FIS. Rnorm is defined in equation (1)

A Heat Power Recognition Using Artificial Neural Networks

A system that consists of a chamber with six SnO₂, was used to recognize the heat power of a given fuel

gas pattern. They detect concentrations of different gases (CO, CH₃, CH₄, etc.). The gas fuel was injected in this chamber with N₂ gas. The concentration ratio of this mixture was adjusted with flow meters. Given a fixed fuel gas flow (3ml/min), the N₂ gas flow was fixed by two flow meters. This gas mixture was applied in the sensor's chamber by the adjustment of another flow meter R7. Because the measurements are in its initial state, few experimental data of only one pattern of gas was acquired with concentration of 5.155ppm of gas fuel. The variation of the N₂ gas flow will simulate the others patterns of gas fuel.

Some measures were taken with sensors steady state values of resistances before and after each injection of the fuel. With these data, a database of data was mounted with six features, corresponding to the sensors values. This database was subdivided in two subsets, a training and a test data sets with equal size.

Synthetic data was used because only few experimental data existed in the moment this paper was made. Three hundred data points was created following gaussian distributions, which corresponds to a hundred for each fuel pattern. Synthetic data of six sensors for a given temperature and pressure values are shown in Fig. 1. Standard deviation and mean of the normalized resistance for the first pattern was obtained with the current available measures. As a first approach, the different patterns were obtained with the same standard deviations of the first class for each sensor, but the second pattern mean was set to a 1/3 ratio of the first pattern, and the mean of the third pattern as a ratio of 1/2.

The data in red are the first fuel pattern with higher heat power value. The second pattern in blue has the least heat power value. The third pattern in green is a fuel pattern with an intermediate heat power value. Fig. 2 shows a scattered plot of a given sensor normalized resistance versus its initial value. The normalized resistance value is defined in equation (1):

$$R_{\text{NORM}} = (R_{\text{INITIAL}} - R_{\text{FINAL}}) / R_{\text{INITIAL}} \quad (1)$$

Two topologies of ANN was tested to recognize the gas fuel patterns: backpropagation and LVQ. To the backpropagation ANN, two train functions was used, traingdx (descendent gradient with adaptive momentum) and trainlm (Levenberg-Marquardt train function), with or without the principal component analysis.

Backpropagation ANN's without PCA had six neurons for the input and hidden layers, and three neurons for the output layer. Backpropagation ANN's with PCA had four neurons for the input and hidden layers, and three neurons for the output layer.

Each ANN was trained 100 times. Stop criterion was mean square error less than 0.01 or error gradient less than 0.001. Data were divided in two subsets, which 150 data points was used to train and 150 data points was used to test the ANN, both with 50 different data points of each fuel gas pattern.

Counts of correct inferences of ANN were made with the test data set. The backpropagation ANN using the training function 'trainlm' was the network that had more corrects inferences, but many times it had some anomalies in the training step. The training of LVQ network was limited to 200 epochs, because it was observed that the mean square error had oscillations after the epoch of number 50.

NeuralWorks was another software used to train the ANN's. The same train and test data sets were used in this program. It had six neurons for the input and hidden neurons, and three neurons for the output layer.

A backpropagation ANN with Delta training rule were used. The network was trained ten times without the "MinMax" table, and more ten times with this table. According to the "MinMax" table records maximums and minimums values of each input data of each network layer. It's used to perform a data pre-processing that applies a scale in layer's input data. Thus it prevents the saturation of the activation function that stops the learning process of the neurons. Training stop criterions were maximum iteration of 50000 epochs, or the root mean square error less than 0.001. The maximum epochs were fixed to ensure that the ANN doesn't suffer "overfitting".

A LVQ neural network was also trained. The learning rule for the firsts 4500 iterations used a conscience factor of 1.0, which encouraged all the neurons in the learning step. For the others 2250 iterations, the training rule did a refinement between patterns. Generalization results of NeuralWorks' ANN's were better than the others networks. Thus, the chosen network to obtain the C code for the neural network implementation in hardware was the backpropagation ANN with "MinMax" table.

Alcohol quality recognition using Artificial Neural Networks

The same system of Taguchi sensors was used to classify the quality of vapor of alcohol fuel. Two patterns of alcohol were considered: good quality (with concentrations between 93% and 98.3%) and the bad quality (others concentrations). The alcohol fuel was injected in the sensors chamber by a syringe. Flow meters devices were replaced by a tube system. A ventilator and an air filter containing a humidity absorber substance were used to clean the chamber. The data acquisition system was the same.

Two ANN's topologies were tested to recognize gas fuel patterns, backpropagation and LVQ, with or without PCA. ANN's without PCA had six neurons for the input and hidden layers, and two neurons for the

output layer. ANN's with PCA had four neurons for the input and hidden layers, and two neurons for the output layer.

Fig. 2 shows a scattered plot with the experimental data of alcohol fuel. Dark blue points represents good alcohol pattern, and pink points represents bad alcohol pattern. Dark blue points inside pink points region can explain why ANN's had some problems of generalization.

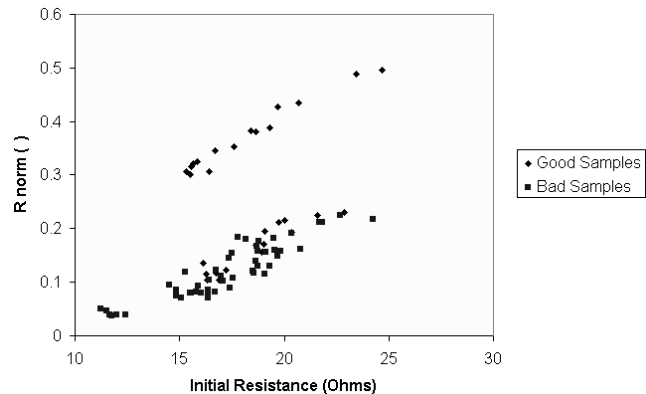


Fig. 2. Scattered plot of normalized resistance variance versus initial resistance obtained by experimental data in sensor 6

Simulations of injection of Alcohol, has lead to creation of a synthetic database. These data was helpful to analyze the effects of pre-processing of data in the training performance of a ANN backpropagation.

Stop criterion of the ANN were root mean square error less than 0.01. DataSculptor software was used to automatically pre-process the data. With the "brief" pre-processing data method, DataSculptor applied a scale in data, and divided the database in test and train subsets. The use of pre-processed data accelerated the ANN training step in a ratio of 0.15.

Hardware implementation of ANN

An ANN trained with NeuralWorks software was implemented in an Analog Devices DSP kit. The NeuralWorks software transformed the trained ANN in a C language program code. No further hardware self-learning steps were considered. Thus, the code generated by NeuralWorks only had mathematical operations, representing the trained neural network.

The initially implemented system was the quality of alcohol fuel recognizer. As a first approach, the test data set was included in the DSP kit program. With a button of the kit, all test data was read, and the results of the trained ANN was shown in two LED's of the kit. The simulation of the trained ANN had the same results of the implemented in the DSP kit.

A dedicated hardware to recognize the quality of alcohol fuel was designed in our laboratory. It has a micro controller, an EEPROM and a FLASH chips. The ANN program was stored in the EEPROM chip, and the synaptic weights were stored in the FLASH chip. As the amount of chip memories was small, the ANN program had to be optimized. This system only process the final and initial response sensor values in steady state. With these values, it calculates and shows the response of the ANN in its LED's.

Fuzzy Inference System

A FIS were used to solve the problem of classification of gas heat power patterns. FIS's rules were extracted from the synthetic data. According to [12], rules are extracted from estimative of data clusters, and each data cluster represents a rule which relates an input region with an output pattern.

Subtractive clustering made the Fuzzy rules extraction. And were obtained a center point for each cluster of each pattern. The features analyzed were the sensor initial resistance and its normalized variation shown in Fig. 3.

Results of FIS without optimizations were not satisfactory. The first 150 samples were applied to FIS sequentially, one by one, during one second for each sample. Among them the first 50 samples were from the first pattern, the next 50 were from second pattern, and the last 50 were from the third pattern. FIS output values ranging from 0 to 0.25 recognizes the first pattern. Output values ranging from 0.25 to 0.75 recognizes the second pattern. And finally, output values ranging from 0.75 to 1.00 recognizes the third pattern.

The behaviors of the ANN's and FIS couldn't have been compared, because the FIS are still in its initial step of design. More accurate differentiation of these two systems would be obtained when more experimental gas fuel data patterns is acquired, and when the FIS is more developed.

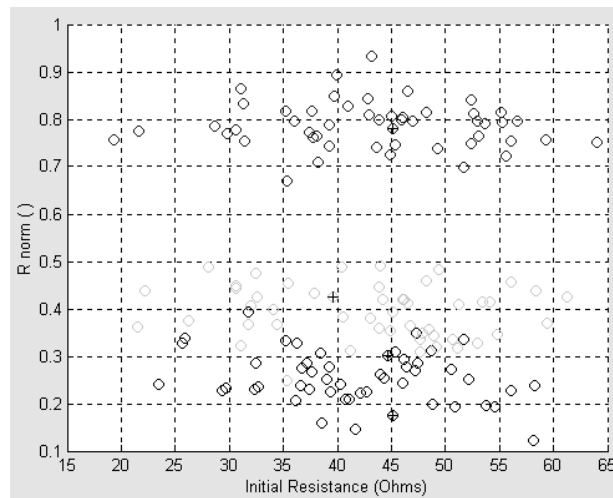


Fig. 3. Scattered plot of normalized resistance variation versus initial resistance of sensor 2 with synthetic data

Conclusions

It has been shown that implementation of an embedded system for classification of heat power of a fuel gas or classification of alcohol fuel is possible. It can be done with an ANN or a FIS. The ANN had a high rate of accurate responses, which shows its capability of generalization.

Principal Component Analysis is an important tool to reduce the input space dimensionality of a system. It can accelerate the training step of a ANN. However, it can prejudice the capability of generalization of the network. Sensors that apparently don't have sufficient sensibility can help the correct classification of a pattern. Also it can be shown that data pre-processing accelerate the training step of an ANN.

FIS can be an alternative solution to the problem. An important advantage is that its model is accessible. ANN's models are difficult to obtain, because each time it is trained, different models are obtained. And the user can't access these models. ANN's behavior is like a non-linear black box.

It was observed that implementation of an ANN in hardware is easier than the implementation a FIS in hardware. It's because the architecture of an ANN is easier than the required architecture to implement a FIS. The most difficult function in a backpropagation ANN is the non linear activate function. FIS requires several mathematical calculi in fuzzification, inference and defuzzification steps.

This work requires the acquisition of more experimental data of different fuel gas patterns, and new ANN training with these new data. With more experimental data, detailed analysis of the sensor's signal in time can be done. Analysis of sensor's several time responses will be possible. Thus, a gas fuel pattern can be faster recognized, improving the performance of the acquisition data system.

The FIS can still be enhanced with a method of automatic optimization of its membership functions. An option is to use a correction function based on gradient descendent.

Hardware implementation of the fuel gas is still in its initial stage. It will be necessary to implement input analogical signal circuits from the sensors, an operator's interface indicating the time in which the embedded system will initiate or terminate a treatment of the electric signals of the sensors, and finally, it will need command software for all this hardware.

References

1. N. Paulsson, E. Larsson, F. Winquist, "Extraction and selection of parameters for evaluation of breath alcohol measurement with an electronic nose", *Sensors and Actuators*, vol. 84, 2000, 187-197.
2. M. A. Drake, P. D. Gerard, J.P. Kleinhenz, W.J. Harper, "Application of an electronic nose to correlate with descriptive sensory analysis of aged Cheddar cheese". *IFT Annual Meeting Book of Abstracts*, New Orleans, Louisiana, p 28, 15C-37.
3. P. E. Keller, L. J. Kangas, L. H. Linden, S. Hashen, R. T. Kouzes, "Electronic noses and their applications", *Institute of Electrical and Electronic Engineers*, October, 1995, 116-119.
4. Y. Yang, P. Yang, X. Wang, "Electronic nose based on SAWS array and its odor identification capability", *Sensors and Actuators B*, vol. 66, 2000, 167-170.

5. J. Gardner, H. Shin, E. Hines, "An electronic nose to diagnose illness", *Sensors and Actuators B*, vol. 70, 2000, 19-24.
6. J. Gardner, H. Shin, E. Hines, C. Dow, An electronic nose system for monitoring the quality of potable water, *Sensors and Actuators B* 69 (2000), 336-341.
7. S. Haykin, *Redes Neurais – Princípios e Prática*, Second Edition, Editora Bookman, Porto Alegre, 27-32.
8. G. Mauris, E. Benoit, L. Folloy, Fuzzy Sensors: An Overview.
9. D.G. Stork; R. O. Duda; P. E. Hart, *Pattern Classification*, Second Edition, Chapter 10.
10. S. V. Kartalopoulos, *Understanding Neural Networks and Fuzzy Logic*, IEEE Press, 1996.
11. R. Ulbig, D. Hoburg, Determination of the calorific value of natural gas by different methods, *Thermochemica Acta* 382 (2000), 27-35.
12. S. L. Chiu. "Extracting Fuzzy Rules from Data for Function Approximation and Pattern Classification". *Fuzzy Information Engineering- A Guided Tour of Applications*. (Eds.: D. Dubois, H. Prade, R. Yager), 1997.

Vitor Hirayama, Walter Jaimes Salcedo, Francisco J. Ramirez-Fernandez ALSTOM BRASIL LTDA, São Paulo, Brazil, E-mail: vitor.hirayama@transport.alstom.com, SIM, LME, USP, Escola Politécnica, Cidade Universitária, São Paulo, SP, Brazil, E-MAIL: JRAMIREZ@LME.USP.BR

УДК 681.327.11

В. П. Майданюк, Х. Ш. Мазін, С. В. Мельник

КОДУВАННЯ ЗОБРАЖЕНЬ З ВИКОРИСТАННЯМ SOFM

Вінницький національний технічний університет, Вінниця

Вступ

Підвищення коефіцієнта ущільнення зображень при збереженні високої якості є пріоритетним напрямком досліджень, які виконуються в цій галузі. Одним із підходів, який забезпечує вирішення цих задач є застосування штучних нейронних мереж.

В науковій літературі розглядаються різні підходи до застосування нейронних мереж при ущільненні зображень [1-3], однак на особливу увагу заслуговують підходи, які ґрунтуються на принципах векторного квантування зображень, оскільки це забезпечує високий коефіцієнт ущільнення при збереженні високої якості відновленого зображення. Ідея векторного квантування дуже проста. Зображення розбивається на квадратні блоки, наприклад 2x2, 4x4 або 8x8. Кожний блок розглядається як вектор в 4-вимірному, 16-вимірному або 64-вимірному просторі. Із цього простору вибирається обмежена кількість векторів, які утворюють кодову книгу, але так, щоб з найбільшою точністю апроксимувати вектори, які вилучаються з вхідного зображення. В канал зв'язку або файл записуються номери векторів з кодової книги, які мають найменшу відстань від векторів, що вилучаються з початкового зображення і сама кодова книга. Оскільки векторів в кодовій книзі значно менше загальної кількості векторів в початковому зображенні, то для представлення номера вектора витрачається менше біт ніж для початкового вектора. За рахунок цього і досягається ущільнення.

Ідеальними для вирішення цих задач є самоорганізуючі нейронні мережі, запропоновані фінським ученим Т. Кохоненом, а саме, самоорганізуючі мережі у вигляді двовимірної карти Кохонена. Карта Кохонена має дві важливі властивості, які використовуються при ущільненні зображень методами векторного квантування. По-перше, вона дуже подібна на інші методи векторного квантування, які застосовують при ущільненні зображень з втратами, а по-друге близьким кластерам вхідних векторів відповідають близько розташовані нейрони, що збільшує ефективність ущільнення без втрат, яке застосовується на останньому етапі ущільнення [3].

Метою даної роботи є розробка та дослідження алгоритмів ущільнення зображень з застосуванням двовимірних карт Кохонена.

Схема ущільнення зображень

Схема ущільнення зображень з використанням карти Кохонена зображена на рис. 1. Після векторизації (перетворення блоків зображення в вектори), виконується векторне квантування з застосуванням карти Кохонена. Вихідні дані векторного квантувача надходять на арифметичний кодер, який виконує кодування зображення без втрат. Декодування виконується в зворотному порядку.

В вихідний файл крім квантованих значень вхідних векторів записується і кодова книга. Але її розмір незначний в порівнянні з вхідним зображенням і це не впливає на коефіцієнт ущільнення. Векторний квантувач це карта Кохонена з розміром 8x8, 16x16 або більшим. Вибір розміру карти Кохонена пояснюється тим, що зображення представляється з точністю 8 бітів на елемент зображення для напівтонових зображень, або 24 біти для кольорового зображення.

Застосування арифметичного кодера на етапі ущільнення без втрат забезпечує найбільший коефіцієнт ущільнення в порівнянні з іншими методами кодування без втрат [5].

Ущільнення включає за два проходи. Під час першого проходу виконується навчання нейронної мережі, а під час другого векторне квантування і ущільнення.

Карта ознак Кохонена

Самоорганізована карта ознак Кохонена (Self-Organizing Feature Map – SOFM) має набір вхідних елементів, кількість яких відповідає розмірності вхідних векторів і набір вихідних елементів, які служать в якості прототипів. Базова архітектура мережі SOFM зображена на рис.2.

Вихідні елементи називаються кластерними елементами. Кластерні елементи або кодові слова розміщуються у вигляді одно- або двовимірного масиву. Звичайно кількість кластерних елементів значно менша в порівнянні з кількістю навчальних зразків, оскільки метою є отримання спрощеної характеристики вхідних даних. Це і дає можливість використання SOFM як векторного квантувача. Після

навчання ця мережа може апроксимувати вектори вхідного простору найкращим способом.

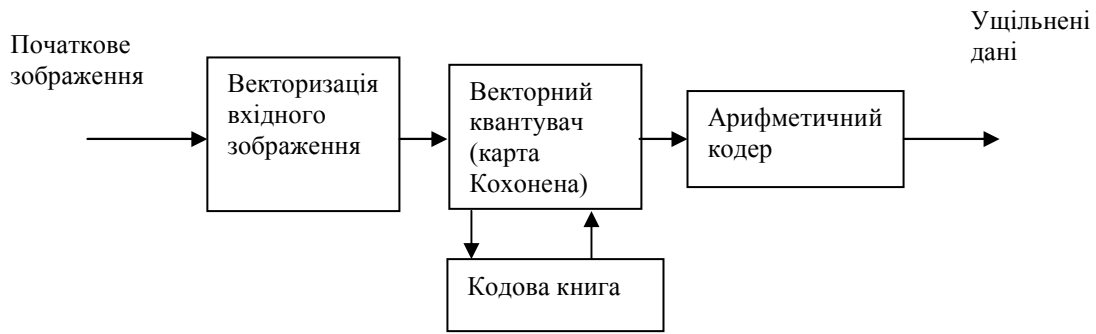


Рис. 1. Загальна схема кодування (кодер)

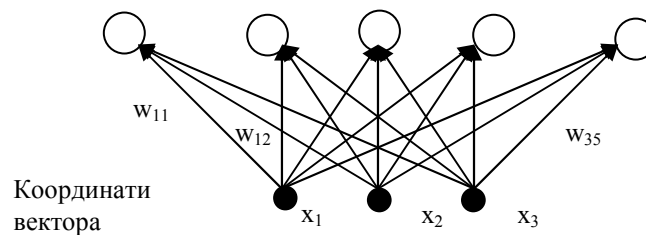


Рис. 2. - Базова архітектура мережі SOFM

Кожний нейрон представляється ваговими коефіцієнтами w_{ij} . Алгоритм навчання мережі такий:

1. Ініціалізувати вагові коефіцієнти випадковими значеннями.
2. Для кожного кластерного елемента обчислити відстань до навчального вектора:

$$d_j = \sum_i (w_{ij} - x_i)^2,$$

3. Знайти кластерний елемент j для якого d_j мінімальне.
4. Для кластерних елементів із круга заданого радіуса з центром в j елементі оновити вагові коефіцієнти згідно з формулою:

$$w_{ij}(n+1) = w_{ij}(n) + \eta(n)[x_i - w_{ij}(n)],$$

де η - норма навчання, x_i – координата навчального вектора.

5. Оновити норму навчання η і радіус за необхідності, і повторити пункти 1-5 для наступного навчального вектора.

Норма навчання з часом змінюється. Вона може, наприклад, мати значення 0,9, а потім змінюватись лінійно до деякого фіксованого значення, наприклад 0,01, після чого залишатись незмінною. Радіус також спочатку вибирається достатньо великим, щоб оновлювалися всі елементи. З часом радіус зменшується і в кінці повинен оновлюватись тільки сам елемент-переможець.

Арифметичне кодування

Основні принципи арифметичного кодування були розроблені наприкінці 70-х років. Арифметичне кодування, так само як і імовірнісні методи, використовує як основу технології ущільнення імовірність появи символу в файлі, однак сам процес арифметичного кодування має принципові відмінності. У результаті арифметичного кодування символна послідовність (рядок) замінюється дійсним числом більшим нуля і меншим одиниці [5].

Процес арифметичного кодування повідомлення полягає в тому, що кожному символу присвоюється інтервал імовірності (range), довжина якого розраховується, виходячи з імовірності його появи в повідомленні.

Перша буква слова одержує інтервал з нижньою межею β_0^l і з верхньою - β_0^h . Нижня межа інтервалу і стає першою значущою цифрою коду. Потім виконується розрахунок границь підінтервалів для кожної наступної букви за такими виразами:

$$\beta_n^l = \beta_{n-1}^l + (\beta_{n-1}^h - \beta_{n-1}^l)P_n^l$$

$$\beta_n^h = \beta_{n-1}^h + (\beta_{n-1}^l - \beta_{n-1}^h)P_n^h,$$

де β^l, β^h – нижня і верхня межі кодового інтервалу, P^l і P^h – нижня і верхня межі інтервалу імовірності для символу.

Арифметичне кодування дозволяє забезпечити високий ступінь ущільнення даних, особливо у випадках, коли зустрічаються дані, де частота виникнення різних символів дуже відрізняється одна від одної. У той же час сама процедура арифметичного кодування вимагає потужних обчислювальних ресурсів, і донедавна цей метод мало застосовувався при кодуванні зображень через повільну роботу алгоритму і, відповідно, істотний час затримки при передачі даних. Однак, враховуючи значне зростання швидкодії апаратних засобів, варто очікувати високих коефіцієнтів ущільнення при його застосуванні для кодування зображень.

Моделювання і результати

При проведенні експериментальних досліджень використовувався комп'ютер з процесором Celeron 700. Вхідне зображення представлено у форматі *.BMP, 24-розряди. Якість відновленого зображення оцінювалась методом експертних оцінок відповідно до 654 рекомендації МККР за такою шкалою:

Якість	Погіршення
Відмінна	Непомітно
Добра	Помітно, але не заважає
Задовільна	Трохи заважає
Погана	Заважає
Дуже погана	Дуже заважає

А також виконувався розрахунок середньоквадратичного відхилення (СКВ) за виразом

$$\sigma = \frac{1}{M * N} \sqrt{(X(m_1, m_2) - \bar{X}(m_1, m_2))^2},$$

де $X(m_1, m_2)$ – значення відліку початкового зображення, а $\bar{X}(m_1, m_2)$ – значення відліку відновленого зображення.

Проводились порівняльні дослідження залежності коефіцієнта ущільнення та якості відновленого зображення від характеристик карти Кохонена, а саме:

- залежність коефіцієнта ущільнення від розміру карти (об'єму кодової книги);
- залежність середньоквадратичного відхилення від розміру карти.

Результати роботи алгоритму для файлу LENA.BMP при 4-вимірних вхідних векторах і розмірах карти від 8x8 до 16x16 подані в табл. 1, а також на рис. 3-4

Аналіз наведених результатів показує, що векторне квантування забезпечує достатньо високі характеристики. Цілий ряд експериментів з різними типами зображень показав, що коефіцієнти ущільнення можуть знаходитися в межах 10 – 30. Для деяких зображень коефіцієнт ущільнення перевершує стандарт JPEG при тій же якості зображення.

Таблиця 1

Експериментальні дані по файлу LENA.BMP

Метод ущільнення	Розмір початкового файла, байт	Розмір стиснутого файла, байт	Коефіцієнт ущільнення	Середньоквадратична помилка	Візуальна оцінка якості
JPEG	192 054	13106	14,7	0,019	Відмінна
Карта Кохонена (16x16)	192054	11740	16,4	0,02	Відмінна
Карта Кохонена (14x14)	192 054	10979	17,5	0,023	Добра
Карта Кохонена (11x11)	192054	9245	20,8	0,026	Добра
Карта Кохонена (8x8)	192054	6954	27,6	0,03	Задовільна



Рис. 3. Зображення після кодування
(карта Кохонена – 16x16)

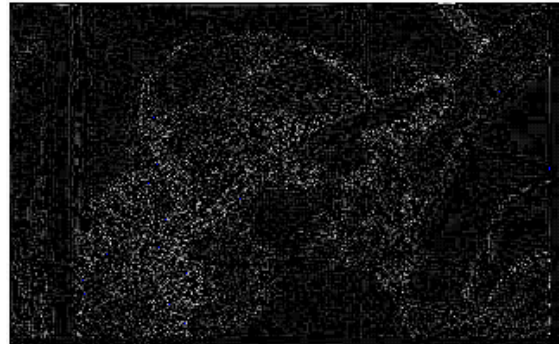


Рис. 4. Різницеве зображення підсилене
в 10 разів

Висновки

1. Проведені дослідження показали, що застосування карти Кохонена є перспективним для виконання ущільнення методами векторного квантування.
2. При векторному квантуванні з використанням карт Кохонена середньо-квадратичне відхилення не в повній мірі відповідає візуальній оцінці якості зображення, для близьких значень відхилення візуальна оцінка якості відновленого зображення різна, хоча в цілому тенденція зберігається – чим менше відхилення тим вища візуальна якість відновленого зображення.
3. Напрямок подальших досліджень пов'язаний з застосування карт Кохонена в комбінації з смуговими методами і двовимірними ортогональними перетвореннями.

Список літератури

1. Прэрт У. Цифровая обработка изображений: Пер. с англ. – М.: Мир, 1982. – Кн.2. – 480 с.
2. Круглов В.В., Борисов В.В. Искусственные нейронные сети. Теория и практика. – М.: Горячая линия – Телеком, 2001. – 382 с.
3. Каллан Р. Основные концепции нейронных сетей.: Пер. с англ. – М.: Вильямс, 2001. – 286 с.
4. Бабак В. П., Ханецький В.С., Шрюфер Е.. Обробка сигналів – К: Либідь, 1996.–392 с.
5. Майданюк В. П. Методи і засоби комп'ютерних інформаційних технологій. Кодування зображень. Вінниця: ВДГУ, 2001. – 63 с.

Майданюк Володимир Павлович, к.т.н., доцент, доцент кафедри програмного забезпечення, Вінницький національний технічний університет, м. Вінниця, Україна, т. 44-04-83, E-Mail-maydan2000@mail.ru.

Мазін Хіллес Шаді, аспірант кафедри лазерної та оптоелектронної техніки, Вінницький національний технічний університет, м. Вінниця, Україна

Мельник Сергій Володимирович, студент 5-го курсу, ФКІ, гр. ІПЗ-00, Вінницький національний технічний університет, Хмельницьке шосе, 95, Вінниця, 21021, Україна, тел.: (0432) 43-78-80

УДК 62.50:658.21

Т. М. Боровська, І. С. Колесник, В. А. Северілов, П. В. Северілов МОДЕЛЮВАННЯ БАНКІВСЬКОЇ СИСТЕМИ

Вінницький національний технічний університет, Вінниця

Постановка проблеми

Ефективна система банків – одна з необхідних умов ефективності національної економіки і добробуту кожного громадянина. Потенційно банки оптимізують розподіл суспільних ресурсів між споживанням, накопиченням і розвитком. Клієнтами банків сьогодні є більша частина населення розвинених країн.

Банк – специфічна система масового обслуговування. Специфіка банківської діяльності у високому рівні конкуренції та „примхливості” клієнтів. Ці два фактори разом породжують високий рівень невизначеності і ризиків.

Загально визнано, що рівень банку визначається інтенсивністю використання і якістю математичних моделей – засобів прогнозування і управління. Життя примушує банки розробляти необхідні робочі моделі. Однак, ці моделі не виходять за межі організації. В програші залишається в першу чергу освіта – майбутніх спеціалістів вчать в основному на словесних та елементарних загальних моделях. Знання основ банківської діяльності потрібні сьогодні практично всім. Парадоксальний аспект проблеми моделювання банків в Україні в тому, що це „нічийна територія”. Для спеціалістів з фінансів це „математика”, для спеціалістів з технічних наук – це „економіка”. Хто повинен розробляти моделі? Фінансист, програміст, математик? Це питання не розв'язується дискусіями. В даній статті подано процес побудови робочої моделі банківської системи на базі теорії управління і системного аналізу.

Аналіз останніх досліджень і публікацій

Обсяг публікацій з банківської справи зростає експоненційно. В Росії і Україні щорічно видаються десятки посібників і підручників – своїх і перекладів з англійської. Однак публікацій з новими і конструктивними результатами небагато. Причини: системні – концентрація і розсіювання знань по джерелах публікацій (закони Бредфорда, Зіпфа), комерційні – ефективні моделі звичайно існують як „ноу-хау”, класифікаційні – помилкова класифікація задач. Рамки стандартної статті не дозволяють провести досить повний аналіз літератури – це окрема тема, для наукології. Переглянута література була поділена на три класи (що можуть перетинатись): а) фундаментальні роботи з моделювання і управління; б) навчальні посібники і підручники, в) дисертації. Особливістю літератури класів б) і в) є вторинність – це поглиблення якоїсь колії, наприклад моделі на базі міжгалузевго балансу, нечіткої логіки та нейронних мереж (особливо популярних в фінансових задачах) та ін. Виділено певні роботи – представники класів: типовий підручник для фінансистів [6, 7], підручник з елементами розробки моделей [8], підручник фундаментального рівня [9], класичний фундаментальний підручник-першоджерело [5]. В дисертаціях з банківської справи (захищених як за економічними так і за технічними спеціальностями) виявлено багато цінних висновків і спостережень. Сьогодні з'являється все більше літератури безпосередньо з моделювання банківських систем. Однак все це не складалося в єдину систему моделей. Тому вирішено не узгоджувати знайдене в літературі, а зробити робочу модель (таку, що виконується в середовищі математичного пакету) з самого початку від твердо встановлених емпіричних законів банківської справи і з використанням методології розробки математичних моделей [4]. Створення моделі не самоціль – модель є інструментом для пошуку оптимальних, адаптивних і просто безпечних управлінь, і тут раціонально йти від фундаментальних джерел [1-3] з теорії управління.

Невирішені частини проблеми

Сьогодні потрібні не стільки нові моделі для нових систем і процесів, скільки раціональні технології розробки математичних моделей здатних відобразити об'єкт, що еволюціонує. Потрібен інший підхід до моделювання, зокрема, робота з моделювання, де формули набрані в текстовому редакторі є прикладом нераціональної технології конструювання моделей.

Постановка задачі

Виділимо з численних тактичних і стратегічних задач банку базову задачу: „дешево брати в оренду гроші” і „дорого здавати”. Виправданням для такої грубої моделі є те, що її потім можна

необмежено надбудовувати і розбудовувати. На рис. 1 подано схему банківської системи, що відповідає базовій задачі банку.

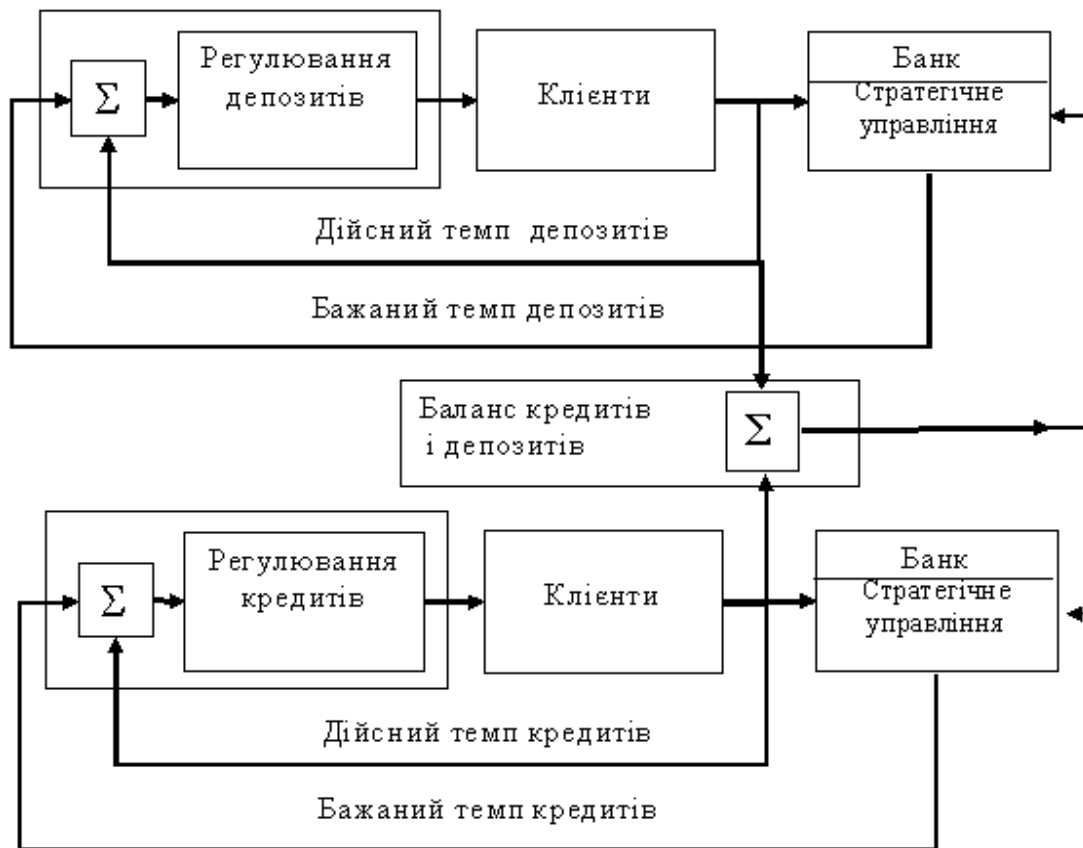


Рис. 1. Схема банківської системи

Ця схема подібна схемі двоканальної системи автоматичного регулювання. Одразу можна бачити одну з головних причин складності – непрогнозований і метаморфозний об'єкт управління – клієнти. Однак, велика кількість і різноманітність клієнтів є скоріше фактором спрощення – в умовах стабільної національної економіки можна усереднювати та агрегувати характеристики клієнтів.

На жаль, в певних умовах поведінка клієнтів буває уніфікованою і детермінованою – це панічне вилучення коштів з банку. Банки, на відміну від виробничих і торгівельних організацій, певним чином зростаються з клієнтами. Очевидно, що треба розглядати банк і клієнтів як єдину – **банківську систему**: банк повинен створювати процвітаючих і надійних клієнтів, клієнти повинні дбати про ефективність і надійність свого банку.

Розробка функціональних моделей

Потоки (темпи) депозитів і кредитів мають певну структуру – розподіли за тривалістю, процентними ставками, надійністю та ін. На першому етапі розглядаємо агреговані усереднені характеристики цих потоків. Працездатну агреговану модель потім неважко модифікувати для урахування структури потоків. Підсистема „депозити” (рис.1) перетворює вхідний темп депозитів у вихідний згідно з правилами і умовами вкладання коштів на депозит і зняття їх. Вхідний темп депозитів залежить від багатьох чинників: ставки депозиту, зростання економіки, темпу інфляції, конкуренції, настроїв і сподівань клієнтів та ін. На першому етапі розглядаємо тільки залежність темпу депозитів від середньої ставки депозиту.

Модель вхідного темпу депозитів

Будемо вважати стан економіки і стан банку (частку фінансового ринку) стабільними. Припускаємо, що кожному рівню ставки депозитів відповідає певний усталений темп депозитів, тобто існує монотонно зростаюча функція ставки депозиту $tdp(rd, A)$, де rd – ставка депозиту, A – потенційний (максимально можливий для даного регіону і періоду часу) темп депозитів.

Реальний темп депозитів тільки в ідеальних умовах досягатиме усталеного рівня. Цей процес

залежить від темпу зростання інформованості клієнтів, наявності прикладу і порад інших клієнтів, а також від ірраціональних та випадкових факторів. Сукупну дію названих факторів на потенційних клієнтів при зміні ставки депозиту задаємо різницеvim рівнянням першого порядку

$$tdvx_{k+1} = tdvx_k + Kd \cdot (tdp(rd, A) - tdvx_k) \cdot dT, \text{ де}$$

$tdvx_k$ - поточний темп депозитів dT - крок обчислень; Kd - коефіцієнт реакції темпу депозитів на зміну ставки. Враховуємо також сезонність та випадкову складову і формуємо відповідний програмний модуль (рис. 2).

$Dpo(rd, dp) =$	<pre> for k ∈ 1..Km kolyv ← 1 + αd·sin(ωM·π·k) - "тренд пропозиції депозитів" tdvx0 ← tdp(rd, Adp)·kolyv - "максимальний темп депозитів" tdvx_{k+1} ← tdvx_k + $\frac{1}{Tdp}$·(tdvx0 - tdvx_k)·dT - "поточний темп" tdvx_{k+1} ← rnorm(1, tdvx_{k+1}, σm)_1 - "випадкова реалізація темпу" </pre>
	$tdvx$

Рис. 2. Модель вхідного темпу депозитів

Модель вихідного темпу депозитів формуємо у вигляді програмного модуля, текст якого подано на рис. 3.

$dpvux(rd, xd, Td) =$	<pre> kzd ← round($\frac{Td}{Tp}$·Np) - "кроків повернення депозитів" for k ∈ 1..Np vupad ← rnd(nakor) + (1 - nakor) - "випадковість" xdd_k ← $\begin{cases} xd(k) & \text{if } (k < kzd) - \text{"повний вхідний темп"} \\ xd(k) + zald_{k-1} & \text{otherwise} \end{cases}$ pas1 ← $\begin{cases} 0 & \text{if } k < kzd - \text{"вихідний темп без процентів"} \\ xdd_{k-kzd} & \text{otherwise} \end{cases}$ pas2 ← pas1·(1 + rd·Td) - "вихідний темп з процентами" pas3 ← pas2·vupadko - "дійсний вихідний темп" zald_k ← pas2 - pas3 - "залишок депозитів" vux^{(k)} ← (stack(xdd_k, pas3, zald_k) - вектор)стану </pre>
	vux

Рис. 3. Модель вихідного темпу депозитів

Входи програмного модуля – ставка депозитів rd , середній термін депозиту і вхідний темп депозитів (фактично це програмний модуль вхідного потоку депозитів), виходи – вихідний темп депозитів без урахування залишків, залишки депозитів.

Модель вхідного темпу кредитів

Як і для темпів депозитів, припускаємо, що кожному рівню ставки кредитів відповідає усталений темп кредитів, тобто існує монотонно спадна функція ставки кредиту $tkr(rk, A)$, де rk – ставка

кредиту, A – потенційний (максимально можливий для даного регіону і періоду часу) темп кредитів.

Реальний темп кредитів тільки в ідеальних умовах досягатиме потенційного рівня - зі зростанням інформованості клієнтів, дії прикладу інших клієнтів, дії ірраціональних та випадкових факторів. Процес реакції потенційних клієнтів на зміну ставки кредиту вважаємо і задаємо різницеvim рівнянням першого порядку

$$tkvx_{k+1} = tkvx_k + Kk \cdot (tkr(rk, A) - tkvx_k) \cdot dT,$$

де $tkvx_k$ – поточний темп кредитів, dT – крок обчислень, Kk – коефіцієнт реакції темпу кредитів на зміну ставки. Враховуємо також сезонність та випадкову складову

Модель вихідного темпу кредитів формуємо на базі вбудованої фінансової функції пакету "виплата боргу рівними частками протягом терміну кредиту:

$$pmt \left(\frac{\text{процент}}{\text{числ_пер_рік}}, \text{число_періодів}, \text{борг_стартовий}, \text{борг_кінцевий} \right).$$

Вихідний потік буде визначатись виразом потрібним інтегралу згортки

$$brgs(k, rk, xk, Tk) := - \sum_{m=1}^k pmt[rk \cdot \Delta t, (Tk \div \Delta t), xk(m), 0] \cdot [(k - m) < (Tk \div \Delta t)]$$

де Δt – крок моделювання, Tk – середня тривалість кредиту, rk – ставка кредиту, $xk(m)$

– значення вхідного темпу кредитів в дискретний момент часу m .

Підсистема управління

Природно, використати математичну модель банківської системи для пошуку ефективних, або хоч безпечних управлінь. На першому етапі поставимо за мету – сформулювати закони управління банківською системою на базі застосування методології і методів теорії управління. Цілі управління для першого етапу – максимізація доходу банку та балансування темпів кредитів і депозитів. Потім на базі працездатної моделі можливо розглянути тонкощі управління – згладжування вхідних темпів кредитів і депозитів, диференціація ставок, урахування прогнозів валютних курсів та економічних показників та ін. Розглядаємо як змінні управління ставки кредитів і депозитів.

Логіка управління

Виділимо "логічні управління": управління балансом і управління для оптимізації доходу. Управління для збалансування темпів кредитів і депозитів повинно одночасно змінювати ставки кредитів і депозитів так, щоб темпи "йшли назустріч". Для першого наближення беремо простіший, пропорційний закон управління балансом:

зміна ставки кредиту $\Delta lrk = K1k \cdot \delta ba$; зміна ставки депозиту $\Delta lrd = -K1d \cdot \delta ba$, де $K1k$, $K1d$ – "коефіцієнти підсилення", δba – різниця темпів депозитів і кредитів (=помилка регулювання). Різні знаки обумовлені тим, що для збільшення темпу кредитів треба зменшувати ставку, а для збільшення темпу депозитів – збільшувати її.

Управління доходом банку відноситься до іншого класу – екстремального: нам потрібно не стабілізувати значення доходу на якомусь заданому рівні, а максимізувати за рахунок вибору значень ставки кредиту – rk . Ставка депозиту прив'язується до ставки кредиту за рахунок умови збалансування темпів кредитів і депозитів. Логіка екстремального управління така: на поточному кроці визначаємо приріст доходу за попередній період, якщо дохід зростає – продовжуємо змінювати управління в тому ж напрямку, що і на попередньому кроці, якщо дохід зменшується – змінюємо управління в протилежному напрямку. Це досить примітивна і наївна логіка "індикаторної поведінки". Можна її вдосконалити введенням фільтрації (не реагувати на випадкові і короткочасні збурення) і прогнозування (реагувати не тільки на зміну доходу, але і на тенденції зміни доходу). Перекладаємо ці міркування на мову рівнянь та алгоритмів. Глянемо також на банківську систему, як на систему „попит-пропозиція”:

- процентна ставка – це ціна грошей,
- залежність "ставка кредиту – темп кредитів" – це крива попиту на гроші,
- залежність "ставка депозиту – темп депозитів" – це крива пропозиції грошей.

На рис. 4 подано постановку і розв'язання задачі про оптимальну різницю ставок – якщо

існують монотонно спадна функція попиту на кредити і монотонно зростаюча функція пропозиції депозитів, то існує оптимальна різниця ставок, що дає максимум процентного доходу. Бачимо, що задача пошуку оптимальної різниці ставок еквівалентна геометричній задачі про вписаний прямокутник з максимальною площею. Можемо бачити (рис. 4) також, що найбільший рівноважний стан досягається при рівних ставках кредиту і депозиту – в точці перетину кривих попиту і пропозиції.

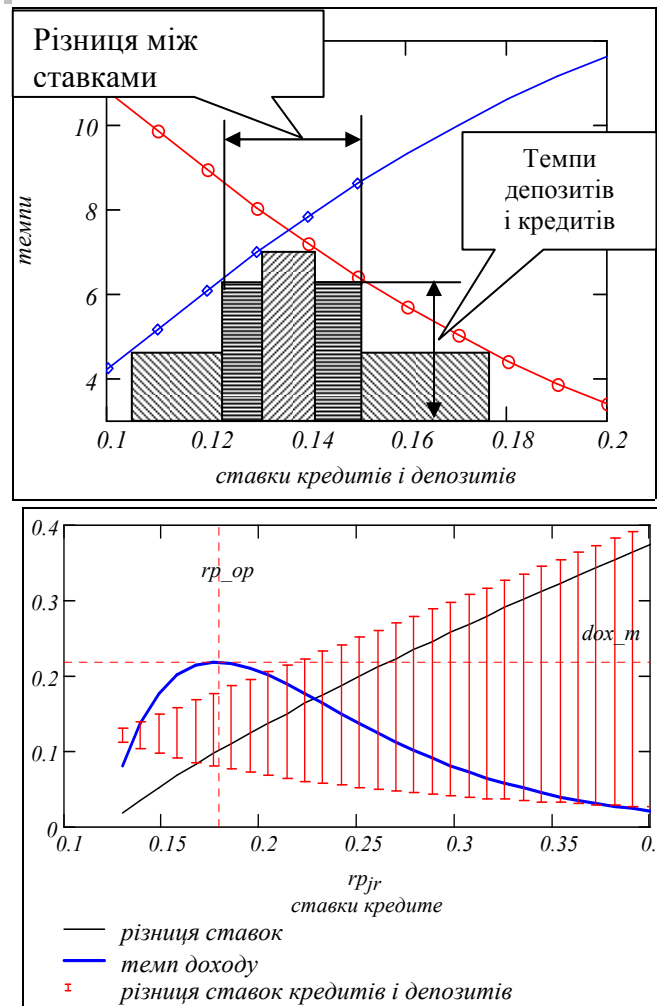


Рис. 4. До визначення оптимальної різниці ставок кредитів і депозитів

Зміна ставок кредитів і визначається дією розглянутих двох регуляторів згідно рівнянням:
 $rK_j = rK_{j-1} + \Delta 1rk + \Delta 2rk$; $rD_j = rD_{j-1} + \Delta 1rd + \Delta 2rd$,

де $\Delta 1rk$, $\Delta 1rd$ – управління від регулятора балансу, $\Delta 2rk$, $\Delta 2rd$ – управління від екстремального регулятора.

Програма моделювання

На рис. 5 подано схему програми моделювання. Програма складається з функціональних модулів, розглянутих вище – вхідних і вихідних темпах кредитів і депозитів, трендах і невизначеностях, алгоритмах управління. За рахунок налаштування і заміни модулів програма дозволяє відобразити будь-яку специфіку функціонування банківської системи. Програма написана на мові програмування математичного пакету. Ця мова максимально наближена до природної мови математики, тому текст програми практично не відрізняється від запису системи рівнянь математичної моделі. Саме тому використовуємо термін „робоча модель”.

Входи програми зібрані у матрицю з вісімнадцяти параметрів (рис. 6), вихід програми – десять дискретних функцій часу: вхідні і вихідні темпи депозитів, баланси "вхід-вихід" кредитів і депозитів, ставки кредитів і депозитів, загальний баланс банку та баланс вхідних темпів кредитів і депозитів.

Інтерфейс програми

Технологія конструювання моделей ставить специфічні вимоги і до тексту програми і до інтерфейсу

введення-виведення. Це можливість доступу до усіх елементів програми, орієнтованість інтерфейсу на обчислювальний експеримент. Вхідні дані для програми вводяться з коментарями, у "змістовному вигляді", а потім збираються у матрицю параметрів MP . В першому рядку матриці зібрані параметри вхідних і вихідних потоків кредитів, в другому – теж саме для депозитів.

$MBU(mp) =$	$dtk \leftarrow$ "крок обчислень" $ome \leftarrow$ "частота сезонних коливань" $kpk \leftarrow$ "кількість кроків повернення кредиту" $kpd \leftarrow$ "кількість кроків повернення депозиту" $rK_1 \leftarrow$ "стартова ставка кредиту" $rD_1 \leftarrow$ "стартова ставка депозиту" "стартові значення темпів кредитів і депозитів:" "стартове значення балансу:" for $j \in Z..Np1$ – "цикл по крокам моделювання" "максимальний рівень вхідного темпу кредитів" "поточне значення вхідного темпу кредитів" "випадкова реалізація вхідного темпу кредитів" "темп повернення поточного кредиту" "вихідний темп кредитів" "баланс кредитів" "максимальний рівень вхідного темпу депозитів" "поточне значення вхідного темпу депозитів" "випадкова реалізація темпу" "алгоритм повернення депозитів" "вихідний темп депозитів" "залишок депозитів" "баланс депозитів" "баланс банку" "алгоритм управління балансом" "алгоритм управління максимізації доходу" "матриця вихідних масивів"
-------------	--

Рис. 5. Схема програми моделювання банківської системи

Результати тестування програми

Програма моделювання (робоча модель) пройшла комплексне тестування для перевірки „синтаксичної” (виявлення банальних помилок – втрачених складових і переплутаних знаків), „семантичної” (перевірка результатів моделювання при нульових та рівних ставках, при однакових термінах кредитів і депозитів) і „прагматичної” (відповідність статистичним даним, можливість управління і настроювання моделі) корекції. Не ставилася ціль детального порівняння результатів моделювання зі статистичними даними – вони є результатом багатьох впливів, типу інфляції та

економічного росту, що не були включені в модель.

Резерв банку $резерв := -3$ Залишкова частка депозиту $vnkr := 0.0$.
Ставки: кредиту $rko := 0.08$, депозиту $rdo := 0.06$, середні терміни:
кредиту $Tko := 5$; депозиту $Tdo := 3$. **Клієнти:**
Кредити: інерційність $Tkr1 := 1.1$; максимальний рівень $po ko := 21$; сезонність
 $ask := 2$ частота $om := 1$, розкид $\sigma k := 0.1$. Параметри попиту
Депозити: інерційність $Tdp1 := 1.1$; максимальний рівень $pod o := 21$; сезонність
 $asd := 2$ частота (=) розкид $\sigma d := 0.1$. Параметри пропозиції
Період моделювання $Tr1 := 30$; число кроків $Np1 := 150$; $k := 1..Np1$;
 $\Delta t := Tr1 \div Np1 \quad \Delta t = 0.2$. **Параметри законів управління:**
регулювання балансу $K1k := 0.003$; $K1d := 0.003$ -натисни;
максимізація доходу $K2k := 0.0003$; $K2d := 0.0003$; $\Delta o := 0.0$.
Матриця параметрів $mp := \begin{pmatrix} rko & Tko & po ko & ask & \sigma k & om & K1k & K2k & Tkr1 \\ rdo & Tdo & pod o & asd & \sigma d & vnkr & K1d & K2d & Tdp1 \end{pmatrix}$

Рис. 6. Модуль вводу параметрів банківської системи

Спочатку була випробувана модель з фіксованими ставками кредитів і депозитів, потім в модель був введений модуль управління з найпростішими законами управління – пропорційними. Параметри законів управління знаходилися експериментально. Задачі аналітичного синтезу управління, побудови адаптивної системи управління, тим більше, – нейронної мережі, не ставилися.

На рис. 7 подано приклади перехідних процесів на відповідних фазових площинах з урахуванням випадкових збурень. Ліворуч – усталені залежності попиту і пропозиції від процентних ставок і фазові траєкторії зміни ставок і темпів в часі. Праворуч – статична залежність темпу доходу від ставки кредиту і фазова траєкторія зміни темпу доходу в часі.

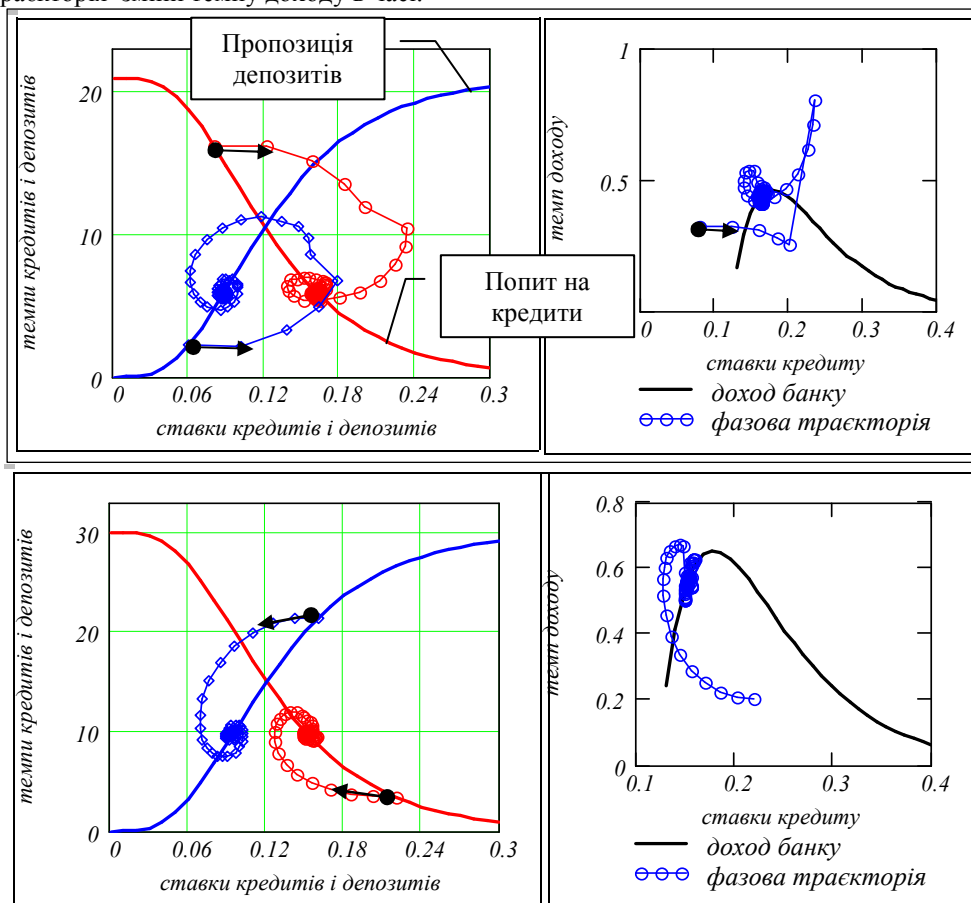


Рис. 7. (продовження) Перехідні процеси на фазовій площині для випадків стартової ставки кредиту нижче і вище оптимальної

Бачимо, що а) підсистема стабілізації балансу вхідних потоків кредитів і депозитів працює задовільно; б) підсистема максимізації доходу є **незадовільною**: не в усіх випадках приводить систему до максимуму, іноді - взагалі робить систему нестійкою. Причини незадовільності підсистеми максимізації: а) підсистеми управління діють сумісно на об'єкти управління, що може приводити до небажаної інтерференції; б) занадто великі збурення і неоптимальні значення коефіцієнтів зворотних зв'язків $K1k, K1d, K2d; K2k$; в) алгоритм пошуку максимуму має органічний порок – коли дохід майже незмінний – припиняється оптимізація.

На рис. 8 подано перехідні процеси в часі для тих же умов, що і на рис. 7. Можемо бачити, що система управління приводить ставки кредиту і депозиту досить точно до оптимального стану (16%, 9%) з початкових станів „малі ставки” (8%, 6%) та „високі ставки” (22%, 8%).

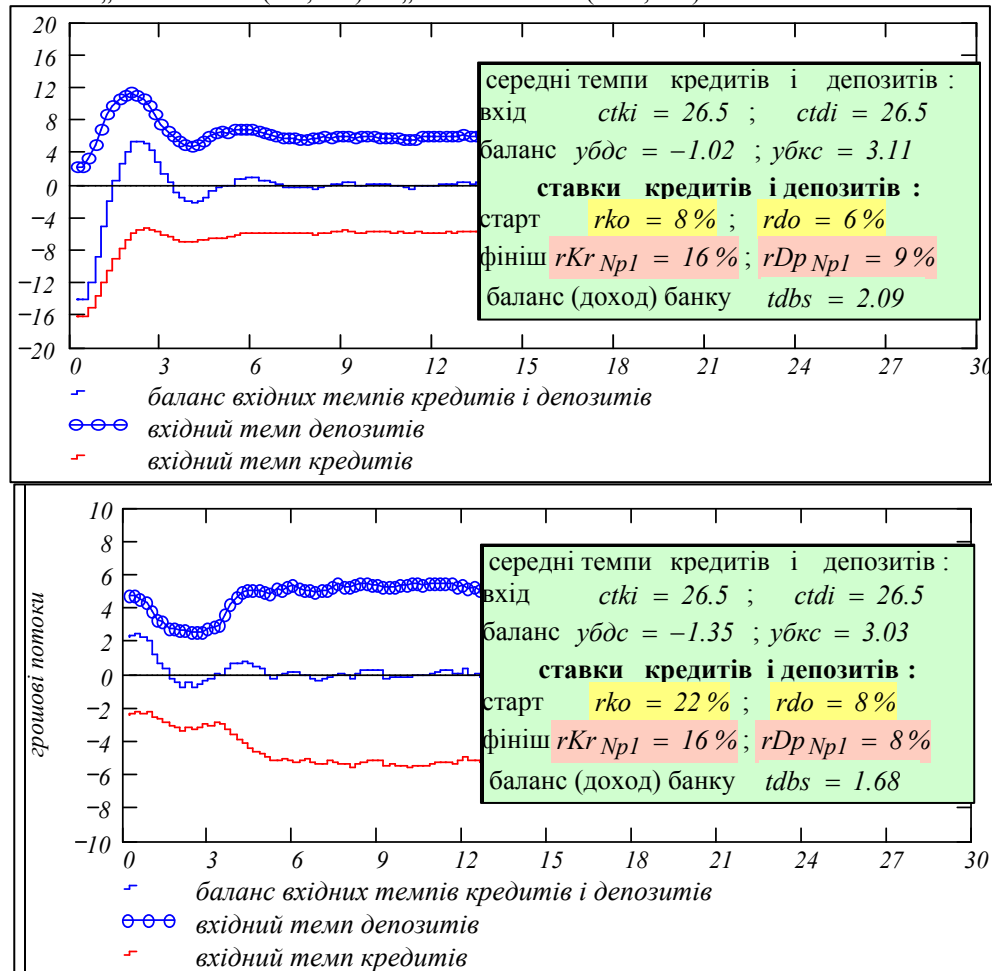


Рис. 8. Перехідні процеси для випадків стартової ставки кредиту нижче і вище оптимальної

Напрямки розвитку моделі банківської системи

Початкова ціль розробки була в створенні максимально спрощеної математичної моделі банку, як системи масового обслуговування. Розробка повинна була дати можливість студентам виконувати експерименти над віртуальною реальністю і так дати початкове розуміння суті банківської справи: джерела нестійкості і виникнення „безпричинних” банкрутств. Робочу модель прийшлося будувати „з нуля”. Навіть найпростіша з реалістичних моделей виявилася досить об'ємною. Однак, ця модель виявилася базовою – вона дозволила відносно легко враховувати і досліджувати різноманітні аспекти банківської діяльності. В даній статті як приклад розширення подано моделі управління банком. Крім цього були розроблені моделі темпів кредитів і депозитів з урахуванням інфляції, конкуренції та панічної поведінки клієнтів. Чисто технічною задачею є модифікація моделі для урахування диференціації ставок кредитів і депозитів а також диференціації клієнтів. Розроблені на базі класичних портфельних задач субмоделі для оптимізації структури активів і пасивів.

Висновки

Побудована базова модель динаміки банківської системи. Модель придатна для розвитку в двох напрямках – описативному, тобто по можливості точному відтворенні поведінки реальних банківських

систем, і прескриптивному, тобто пошуку режимів функціонування і методів управління, що не спостерігалися в реальності, але є бажаними і можливим для реалізації. На конкретному прикладі показано виникнення та існування науковоізолюваних областей на стиках традиційних наук. Моделювання банківської системи не є класичною задачею фінансів, а задачею системного аналізу та теорії управління.

Список літератури:

1. Беллман Р., Гликсберг И., Гросс О. Некоторые вопросы математической теории управления. — М.: Издат. иностр. литер., 1962. — 233с.
2. Беллман Р. Процессы регулирования с адаптацией. — М.: Наука, 1964. — 317с.
3. Цыпкин Я.З. Адаптация и обучение в автоматических системах. — М.: Наука, 1968. — 400с.
4. Пешель М. Моделирование сигналов и систем. — М.: Мир, 1981.-286с.
5. Самуельсон П. Економіка: - Львів: Світ, 1996. — 433с.
6. Банки и банковское дело; Под ред. Балабанова И.Т. — Санкт-Петербург: Питер, 2001— 255с.
7. Банківська справа: Навч. Посібник; За ред. Тиркала Р.І. — Тернопіль: Карт-бланш, 2001. —314с.
8. Васюренко О.В. Банківський менеджмент: Посібник. — К.: вид.центр „Академія”, 2001.— 320с.
9. Вітлінський В.В. Моделювання економіки: Навч. посібник: - К.: КНЕУ, 2003. — 408с.
10. Кизим Н.А. Моделирование банкротства коммерческих банков. — М.: Финансы, 2002. — 224с.

Боровська Таїсія Миколаївна, к.т.н., доцент кафедри комп'ютерних систем управління, Вінницький національний технічний університет, Вінниця.

Колесник Ірина Сергіївна, здобувач кафедри комп'ютерних систем управління, Вінницький національний технічний університет, Вінниця.

Северілов Віктор Андрійович, к.т.н., доцент кафедри інформаційних технологій, Відкритий міжнародний університет розвитку людини, Вінниця.

Северілов Павел Вікторович, директор пп „Центр Мультимедиа”

УДК 007+681.3

О. М. Хошаба, А. В. Савін, В. Е. Корбут

РОЗВ'ЯЗАННЯ ДЕЯКИХ ЗАДАЧ ПОБУДОВИ МАТЕМАТИЧНОЇ МОДЕЛІ ДЛЯ РУХОМИХ ОБ'ЄКТІВ

Вінницький національний технічний університет, Вінниця

Вступ

В теперішній час важливими є рішення задач побудови інтелектуальних систем, які працюють у реальному часі. В цьому випадку є дослідження поведінки окремих агентів мультиагентної системи яка працює як автономна структура. Тому, розвиток фундаментальних методів дослідження дозволить прискорити розробки сучасних інтелектуальних систем реального часу. До таких досліджень відноситься моделювання систем рухомих об'єктів.

Постановка задачі

На основі експериментальних даних руху об'єктів (радіокерованих машинок) визначити пряму та обернену функцію основних параметрів до яких відноситься: величина сигналу, довжина шляху, час руху об'єкта, його середня швидкість. Визначення цих параметрів повинно задаватись функціями:

- для прямих функцій:

$func1(s, coef);$

значення які повертаються функцією:

x_Z, t_Z

$func2(s, t_T, coef);$

значення які повертаються функцією:

$x_T, v_{СЕР}$

- для обернених функцій:

$func3(x_Z, coef);$

значення які повертаються функцією:

s, t_Z

$func4(t_Z, coef);$

значення які повертаються функцією:

s, x_Z

де s – величина одиничного сигналу,

x_Z – загальна довжина шляху рухомого об'єкта,

t_Z – загальне значення часу,

x_T – довжина шляху рухомого об'єкта у визначеній точці,

t_T – значення часу рухомого об'єкта у визначеній точці,

$v_{СЕР}$ – значення середньої швидкості рухомого об'єкта у визначеній точці,

$coef$ - коефіцієнт, який визначається в ході тестування рухомого об'єкта.

В зв'язку з тим, що система буде працювати у реальному режимі часу, розрахунки з визначення основних характеристик руху повинні мати високу швидкодню.

Мета дослідження

Побудова математичної моделі рухомого об'єкта (радіокерованими машинками).

Рішення задачі

Вирішення задачі складається з розв'язання цілої низки математичних та технічних задач за такими напрямками: розпізнавання образів, вибір оптимальної стратегії руху, побудови математичних моделей з виконання дій по переміщенню об'єкта. В даній статті розглянута проблема визначення швидкості руху та довжини переміщення об'єкта при поданні одиничного сигналу.

Одним з вирішенням питань даної проблеми є спроба співвідношення математичної моделі рухомого об'єкта до механічних законів фізики, наприклад, другого закону Ньютона. Однак, у зв'язку з тим, що „сила” одиничного сигналу залежить від заряду акумулятора рухомого об'єкта, використання

цієї залежності становить обчислення основних характеристик руху за допомогою другого закону Ньютона не доречним. Тому, вирішення функціональних залежностей руху дослідних об'єктів може бути поділено на розв'язок двох задач, коли одиничний сигнал подається у положенні „зупинка” та „рух”. Перший випадок передбачає виконання таких дій:

- 1) формування групи сигналів, які характеризують дію „не можливий рух”;
- 2) формування групи сигналів, які характеризують дію „рух – зупинка”;
- 3) формування групи сигналів, які характеризують дію „рух – максимальна швидкість – зупинка”;
- 4) визначення функціональних залежностей у групі дій „рух – зупинка”;
- 5) визначення функціональних та кореляційних залежностей у групі дій „рух – максимальна швидкість – зупинка”;
- 6) визначення особливостей руху „вперед”, „назад”, „ліворуч”, „праворуч”.

У другому випадку виконується рух коли швидкість об'єкта дорівнює не менш 0,01 м за 0,1 с.

Для вирішення задачі яка сформульована в розділі „Постановка задачі” були проведені експериментальні дослідження руху дослідного об'єкту (табл. 1) та визначені функціональні залежності між основними показниками (табл. 2, 3).

Таблиця 1

Значення експериментальних досліджень руху дослідного об'єкта

Величина одиничного сигналу	Довжина проміжку часу, 0,1 с			Довжина руху, м			Середня швидкість руху, м/0.1с		
	t_1	t_2	t_3	t_1	t_2	t_3	t_1	t_2	t_3
350	0,7	0	1,0	0,41	0	0,29	0,59	0	0,29
400	1	0	1,0	0,59	0	0,29	0,59	0	0,29
450	0,3	0,7	1,3	0,18	0,49	0,51	0,60	0,70	0,39
500	0,3	0,9	1,3	0,18	0,64	0,51	0,60	0,71	0,39
550	0,3	1,1	1,3	0,18	0,79	0,51	0,60	0,72	0,39
600	0,3	1,2	1,3	0,18	0,88	0,51	0,60	0,73	0,39
650	0,3	1,3	1,3	0,18	0,96	0,51	0,60	0,74	0,39
700	0,3	1,4	1,3	0,18	1,04	0,51	0,60	0,74	0,39

Формування групи сигналів, які характеризують дію „рух – зупинка” мають динаміку зміни величин показників, які зображені на рис. 1 штриховими лініями.

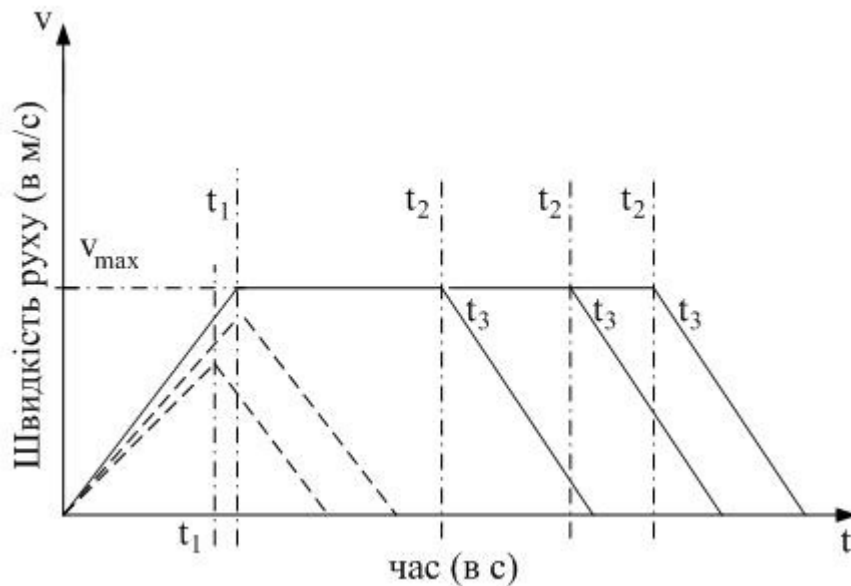


Рис. 1. Часові траєкторії руху дослідних об'єктів за різними величинами одиничного сигналу

На рис. 1 штриховими лініями показано сигнали, які мали динаміку зміни величин показників на відрізках часу t_1 та t_3 . Ця група сигналів відрізнялась від групи сигналів, які характеризують дію „рух – максимальна швидкість – зупинка” відсутністю відрізка часу t_2 який визначав би максимальну швидкість об'єкту за певний час.

Формування групи сигналів, які характеризують дію „рух – максимальна швидкість – зупинка” мають іншу динаміку зміни величин показників, які зображені на рис. 2. Ця інша динаміка полягає у досягненні максимальної швидкості руху дослідним об'єктом протягом деякого часу. В нашому випадку, до цієї групи належали сигнали, величина яких дорівнювала або була більшою ніж 450 одиниць (табл. 1).

За допомогою апроксимації за методом найменших квадратів для двох дослідних груп були визначені функціональні залежності між значенням величини одиничного сигналу та часом у формі:

$$y = ax^2 + bx + c \text{ (табл. 2, 3).}$$

Таблиця 2

Функціональні залежності між основними показниками для величини довжини руху

Величина одиничного сигналу (в од.)	Коефіцієнти квадратичного рівняння		
	a	b	c
350	-20	77	-3,6
400	-16	79	-5,1
450	-13	85	-5,1
500	-13	89	-6,1
550	-12	91	-7,1
600	-12	94	-8,3
650	-12	95	-8,9
700	-12	96	-9,5

Таблиця 3

Функціональні залежності між основними показниками для величини середньої швидкості

Величина одиничного сигналу (в од.)	Коефіцієнти квадратичного рівняння		
	a	b	c
350	-0,19	0,015	0,60
400	-0,21	0,17	0,56
450	-0,15	0,14	0,64
500	-0,14	0,15	0,65
550	-0,13	0,17	0,65
600	-0,14	0,19	0,66
650	-0,13	0,20	0,65
700	-0,13	0,22	0,65

На основі даних, які показані в табл. 2 та 3 з'явилась можливість побудови прямих функцій $funct1$, $funct2$ (розділ „Постановка задачі”). Так, для функції $funct1(s, coef)$ будемо мати:

$$x_Z = -0,00066s^2 + 0,98s - 200 \text{ (см);}$$

$$t_Z = 0,0036s + 0,56 \text{ (с);}$$

За замовчуванням, значення коефіцієнта $coef$ дорівнює одиниці. Однак, при необхідності тестів це значення змінюється і результати x_Z та t_Z помножуються на значення коефіцієнта.

Визначення змінної x_T для функції $funct2(s, t_T, coef)$ виконується у два етапи. Перший етап полягає у визначенні загальної кривої, параметром якої є s , а невідома величина – довжина шляху. Тому, використовується формула:

$$kr(s) = ax^2 + bx + c,$$

де коефіцієнти квадратичного рівняння знаходяться по формулам:

$$a = 0,019s - 24;$$

$$b = 0,058s + 58;$$

$$c = -0,017s + 2,2.$$

На другому етапі по отриманим значенням коефіцієнтів a , b та c визначаємо довжину шляху у заданій точці:

$$x_T = kr(s);$$

Визначення значення величини середньої швидкості в заданій точці також виконується у два етапи як у попередньому випадку. Коефіцієнти a , b та c будуть дорівнювати:

$$a = 0,002s - 0,26;$$

$$b = 0,00042s - 0,063;$$

$$c = 0,0002s + 0,52;$$

$$v_{CEP} = vr(s);$$

Для оберненої функції $funct3(x_Z, coef)$ отримали такі результати обчислень:

$$s = 0,025x^2 - 2,9x + 440 \text{ (од.)};$$

$$t_Z = 0,0036s + 0,56 \text{ (с)};$$

Для оберненої функції $funct4(t_Z, coef)$ отримали такі результати обчислень:

$$s = 160t^2 - 480t + 730 \text{ (од.)};$$

$$x_Z = -0,00066s^2 + 0,98s - 200 \text{ (см)};$$

Для визначення особливостей виконання руху праворуч, ліворуч або назад необхідно використовувати коефіцієнти до визначених функцій, які отримуються в ході експериментів та відповідно дорівнюють 0,9;0,9;0,8. Такі значення коефіцієнтів є доцільними в зв'язку з тим, що модель повинна працювати у режимі реального часу.

Висновки

Для розв'язання задачі побудови математичної моделі для рухомих об'єктів необхідно:

- визначити групові характеристики рухомого об'єкта;
- за допомогою методів апроксимації провести розрахунки з аналітичних залежностей .

Побудовані таким чином математичні моделі адекватно описують рухомі об'єкти та використовуються в навчальному процесі для дослідження поведінки інтелектуальних агентів у мультиагентних системах.

Хошаба Олександр Мирославович, к.т.н., доцент, доцент кафедри інтелектуальних систем, Вінницький національний технічний університет, вул. Хм. шосе 95, Вінниця, Україна, 21021, Тел.: 8(0432) 44-04-90, E-Mail- Alexander@Hoshaba.org

Савін Андрій Володимирович, студент кафедри інтелектуальних систем, Вінницький національний технічний університет, вул. Хм. шосе 95, Вінниця, Україна, 21021

Корбут Володимир Едуардович, студент кафедри інтелектуальних систем, Вінницький національний технічний університет, вул. Хм. шосе 95, Вінниця, Україна, 21021

УДК 683.32

Д. А. Зубов

ВЫБОР АЛГОРИТМА ОПТИМАЛЬНОГО УПРАВЛЕНИЯ НЕЛИНЕЙНЫМИ ДВУХКАНАЛЬНЫМИ ОБЪЕКТАМИ УГЛЕОБОГАЩЕНИЯ

Восточноукраинский национальный университет имени Владимира Даля, Україна

Введение

Выбор эффективного алгоритма оптимального управления является одним из базовых этапов при построении систем управления. Решение данной задачи в многомерных системах автоматического управления (МСАУ) часто усложняется мультипликативным и аддитивным влиянием каналов управления, нелинейными зависимостями диофантового характера (количество входных переменных больше выходных [1, 2] – МСАУ не р-канонической структуры [3]), нестационарностью и стохастичностью характеризующих процесс параметров др. Анализ литературных источников (например, [3, 4]) и информационных ресурсов Internet показывает отсутствие в настоящее время единственного универсального подхода автоматического управления такими объектами. Для повышения эффективности функционирования МСАУ представляется целесообразным выделить следующие приоритетные направления исследований:

1. Аналитический синтез закона регулирования (уменьшение количества неизвестных переменных методом Диофанта [1, 2] или другими эквивалентными преобразованиями, вычисление оптимальных управляющих воздействий при линеаризации нелинейной части [3], др.).

2. Комбинаторные алгоритмы реального времени поиска глобального экстремума [5, 6].

3. Методы многомерного поиска управляющих воздействий (градиентный, Хука-Дживса, конфигураций Розенброка др. [7]).

Целью данной статьи является исследование первых двух направлений для управления технологическими процессами (ТП) углеобогащительной фабрики (УОФ), которые имеют два управляющих воздействия (для отсадки это высота и степень разрыхленности породной постели, для флотации – удельные расходы реагентов собирателя и пенообразователя, для тяжелых сред – плотность и вязкость магнетитовой суспензии [5]) и одну выходную координату (зольность концентрата ТП). Данная задача является актуальной для предметной области автоматизации ТП УОФ в связи с тем, что ранее управление ТП как динамическими, нелинейными, двухканальными, нестационарными, стохастическими объектами с большим запаздыванием и диофантовым характером зависимостей не рассматривалось.

Модель ТП УОФ

Двухканальная дискретная модель приведенной непрерывной части ТП УОФ с экстраполятором нулевого порядка имеет вид [5]:

$$\begin{aligned} y_1[n] &= b_{1,1}[n]u_1[n-d_1[n]-1] + b_{1,2}[n]u_1[n-d_1[n]-2] - a_{1,1}[n]y_1[n-1] - a_{1,2}[n]y_1[n-2] + v_{1,o}[n]; \\ y_2[n] &= b_{2,1}[n]u_2[n-d_2[n]-1] + b_{2,2}[n]u_2[n-d_2[n]-2] - a_{2,1}[n]y_2[n-1] - a_{2,2}[n]y_2[n-2] + v_{2,o}[n]; \\ y[n] &= c_0[n] + c_1[n]y_1[n] + c_2[n]y_2[n] + c_3[n]y_1^2[n] + c_4[n]y_2^2[n] + c_5[n]y_1[n]y_2[n] + v_H[n], \end{aligned} \quad (1)$$

где $y_1[n], y_2[n], u_1[n], u_2[n] \in \mathbb{R}$ – выходная координата и управляющее воздействие соответственно по первому и второму каналам управления в дискретный момент времени n (\mathbb{R} – множество вещественных чисел); $v_{1,o}[n], v_{2,o}[n] \in \mathbb{R}$ – случайный нормальнораспределенный шум соответственно по первому и второму каналам управления с нулевым математическим ожиданием и ограниченными среднеквадратическими отклонениями $\sigma_{1,o}[n], \sigma_{2,o}[n] \in \mathbb{R}$; $v_H[n] \in \mathbb{R}$ – случайный нормальнораспределенный шум наблюдения с нулевым математическим ожиданием и ограниченным среднеквадратическим отклонением $\sigma_H[n] \in \mathbb{R}$; $d_1[n], d_2[n] \in \mathbb{Z}$ – нестационарные дискретные запаздывания соответственно по первому и второму каналам управления, $d_1[n], d_2[n] > 0$ (\mathbb{Z} – множество целых чисел); $c_0[n], c_1[n], c_2[n], c_3[n], c_4[n], c_5[n] \in \mathbb{R}$ – коэффициенты квадратичного полинома, аппроксимирующего нелинейную часть модели; $a_{1,1}[n], a_{1,2}[n], b_{1,1}[n], b_{1,2}[n], a_{2,1}[n], a_{2,2}[n], b_{2,1}[n], b_{2,2}[n] \in \mathbb{R}$ – коэффициенты дискретных передаточных функций линейной части модели соответственно по первому и второму каналам управления.

Начальные условия:

© Д. А. Зубов, 2004

$$y_0 = y_0^0; y_{-1} = y_{-1}^0; u_{1,s} = u_{1,s}^0; u_{2,s} = u_{2,s}^0;$$

где $u_{1,s}^0, u_{2,s}^0 \in \mathbb{R}$ – положительные константы; $s = (-d_{max}-1), \dots, (-1), s \in \mathbb{Z}$; $d_{max} \in \mathbb{Z}$ – максимальное запаздывание.

Ограничения на управляющие и возмущающие воздействия, запаздывания модели (1) наряду с ограничением на выходную координату [5]:

$$\begin{aligned} u_{1,\min} \leq u_1[n] \leq u_{1,\max}; u_{2,\min} \leq u_2[n] \leq u_{2,\max}; d_{\min} \leq d_1[n] \leq d_{\max}; u_1[n] - u_1[n-1] = v_1 \cdot u_{1,\min}^*; \\ u_2[n] - u_2[n-1] = v_2 \cdot u_{2,\min}^*; k_\sigma (\sigma_{1,n}[n] + \sigma_{1,o}[n] + \sigma_{2,o}[n]) < \sigma_{\max}; d_{\min} \leq d_2[n] \leq d_{\max}, \end{aligned}$$

где $u_{1,\min}^*, u_{2,\min}^* \in \mathbb{R}$ – минимальные дискретные шаги изменения управляющего воздействия соответственно по первому и второму каналу; $v_1, v_2 \in \mathbb{Z}$ – множители дискретных шагов изменения управляющего воздействия соответственно по первому и второму каналу; $u_{1,\min}, u_{2,\min} \in \mathbb{R}$ – неотрицательные константы; $y_{\min}, y_{\max}, u_{1,\max}, u_{2,\max}, \sigma_{\max} \in \mathbb{R}$ – положительные константы; $k_\sigma \in \mathbb{R}$ – эмпирический коэффициент, отражающий влияние $\sigma_{1,o,n}, \sigma_{2,o,n}, \sigma_{1,n}$ на максимальное среднеквадратическое отклонение выходной координаты σ_{\max} .

Для оценивания параметров модели (1) выходная координата фильтруется методом стохастической аппроксимации:

$$y^{(1)}[n] = y^{(1)}[n-1] + k_a (y[n] - y^{(1)}[n-1]), \quad (2)$$

где $y^{(1)}[n]$ – аппроксимируемое значение выходной координаты; $k_a \in \mathbb{R}$ – в общем случае переменный коэффициент, удовлетворяющий условиям теоремы Дворецкого.

Для фильтрации зашумленной выходной координаты, повышения информативности обучающей выборки данных и обработки аварийных ситуаций предлагается следующая эвристическая пятиуровневая модель представления данных для расчета коэффициентов аппроксимирующей модели [5]:

1. В каждый дискретный момент времени непосредственно снимаемая с датчиков информация о выходной координате – $y[n]$.
2. В каждый дискретный момент времени фильтрация $y[n]$ методом стохастической аппроксимации по уравнению (2) – $y^{(1)}[n]$.
3. В каждый дискретный момент времени вычисление элемента трехмерного массива $y^{(2)}$ по формуле (при запуске программы все элементы обнуляются)

$$y^{(2)} \left[\text{Trunc} \left(\frac{N_e^* (y^{(1)}[n] - y_{\min} - \sigma_{\max})}{y_{\max} - y_{\min} - 2\sigma_{\max}} \right); \text{Trunc} \left(\frac{N_e^* (y^{(1)}[n-1] - y_{\min} - \sigma_{\max})}{y_{\max} - y_{\min} - 2\sigma_{\max}} \right); \text{Trunc} \left(\frac{N_e^* (y^{(1)}[n-2] - y_{\min} - \sigma_{\max})}{y_{\max} - y_{\min} - 2\sigma_{\max}} \right) \right] = n,$$

где N_e^* – количество интервалов длиной $(y_{\max} - y_{\min} - 2\sigma_{\max})/N_e^*$ в возможном диапазоне $(y_{\max} + \sigma_{\max}; y_{\min} - \sigma_{\max})$ изменения выходной координаты; $\text{Trunc}(\cdot)$ – выделение целой части вещественного аргумента.

С учетом интервала возможного изменения выходной координаты и максимального среднеквадратического отклонения шумов объекта и наблюдения σ_{\max} элементы трехмерного массива $y^{(2)}$ формируются данными, удовлетворяющими условию:

$$(y_{\min} + \sigma_{\max}) < y^{(1)}[n] < (y_{\max} - \sigma_{\max}).$$

4. При окончании цикла длиной N_e формирование новой обучающей выборки данных $y^{(3)}_m$ длиной N_e из ненулевых элементов трехмерного массива $y^{(2)}$ (N_e соответственно равно количеству ненулевых элементов трехмерного массива $y^{(2)}$; $m = 1, \dots, N_e$).

5. Для обработки ситуации “зависания” выходной координаты (при исследованиях – среднеквадратическое отклонение на интервале длиной N_e меньше $0,01y_3$, y_3 – уставка выходной координаты) вне допустимого диапазона (при исследованиях – $y_3 \neq 0,05y_3$) из-за “плохой” обусловленности обучающей выборки данных в $y^{(3)}_m$ копируются данные из массива $y^{(4)}$, который формируется на базе обработанной “хорошо” обусловленной информации (при исследованиях в $y^{(4)}$ заносились данные из начальной обучающей выборки).

Прогнозирование выходной координаты путем ее пошагового экстраполирования не представляется возможным при разных значениях запаздывания в каналах управления. Поэтому

запаздывание допускається однаковим для всіх каналів і апроксимуюча модель адекватно структурі (1) має вигляд:

$$y^{(1)}[n] = c_0^*[n] + c_1^*[n]y^{(1)}[n-1] + c_2^*[n]y^{(1)}[n-2] + c_3^*[n]u_1[n-1-d^*[n]] + c_4^*[n]u_2[n-1-d^*[n]] + c_5^*[n]u_1[n-2-d^*[n]] + c_6^*[n]u_2[n-2-d^*[n]] + c_7^*[n](u_1[n-1-d^*[n]])^2 + c_8^*[n](u_2[n-1-d^*[n]])^2 + c_9^*[n](u_1[n-2-d^*[n]])^2 + c_{10}^*[n](u_2[n-2-d^*[n]])^2 + c_{11}^*[n]u_1[n-1-d^*[n]]u_2[n-1-d^*[n]] + c_{12}^*[n]u_1[n-1-d^*[n]]u_2[n-2-d^*[n]] + c_{13}^*[n]u_1[n-2-d^*[n]]u_2[n-1-d^*[n]] + c_{14}^*[n]u_1[n-2-d^*[n]]u_2[n-2-d^*[n]], \quad (3)$$

де $c_{0[n]}^* \neq c_{14[n]}^* \in \mathbb{R}$, $d^*[n] \in \mathbb{Z}$ – константи, апроксимуються методом найменших квадратів (МНК) на навчаючій вибірці даних довжиною N_e для дискретного моменту часу n .

Уменьшение количества переменных уравнения (3) методом Диофанта

Аналіз формули (3) показує мультиплікативне і адитивне впливання каналів управління, квадратичну залежність аргументів. Тому цілорозумним представляється застосування методу Диофанта [1, 2] для зменшення кількості невідомих шляхом отримання їх лінійних залежностей від однієї обобщенної змінної, що дозволить, наприклад, скоротити на порядок обсяг вирахувань при комбінаторному алгоритмі.

Так як управляючі впливання і вихідна координата належать полю раціональних чисел, то існує деяке раціональне рішення. Запишемо (3) в вигляді полінома Колмогорова-Габбора другого порядку з двома аргументами:

$$f_2(u_1[n], u_2[n]) = a_0 + a_1 \cdot u_1[n] + a_2 \cdot u_2[n] + a_3 \cdot u_1^2[n] + a_4 \cdot u_2^2[n] + a_5 \cdot u_1[n] \cdot u_2[n] = 0, \quad (4)$$

де $a_1 = c_3^* + c_{12}^* \cdot u_2[n-1]$; $a_2 = c_4^* + c_{13}^* \cdot u_1[n-1]$; $a_3 = c_7^*$; $a_4 = c_8^*$; $a_5 = c_{13}^*$; $a_0 = -y_3 + c_{0[n]}^* + c_{11}^* \cdot y^{(1)}[n+d^*[n]] + c_{12}^* \cdot y^{(1)}[n+d^*[n]-1] + c_{13}^* \cdot u_1[n-1] + c_{14}^* \cdot u_2[n-1] + c_{15}^* \cdot u_1^2[n-1] + c_{16}^* \cdot u_2^2[n-1] + c_{17}^* \cdot u_1[n-1] \cdot u_2[n-1]$.

Допустимо, що (4) має деяке раціональне рішення (u_1^*, u_2^*) . По методу Диофанта необхідно зробити підстановку:

$$u_1[n] = u_1^* + t; \quad u_2[n] = u_2^* + k \cdot t, \quad (5)$$

де k – обобщенна змінна; t знаходиться з формул:

$$t = -(A(u_1^*, u_2^*) + k \cdot B(u_1^*, u_2^*)) / C(u_1^*, u_2^*, k); \quad (6)$$

$$f_2(u_1^* + t, u_2^* + k \cdot t) = f_2(u_1^*, u_2^*) + t \cdot A(u_1^*, u_2^*) + k \cdot t \cdot B(u_1^*, u_2^*) + t^2 \cdot C(u_1^*, u_2^*, k) = 0. \quad (7)$$

Сравнивая (4) и (7), получаем:

$$A(u_1^*, u_2^*) = a_1 + 2 \cdot a_3 \cdot u_1^* + a_5 \cdot u_2^*; \quad B(u_1^*, u_2^*) = a_2 + 2 \cdot a_4 \cdot u_2^* + a_5 \cdot u_1^*; \quad C(u_1^*, u_2^*, k) = a_3 + 2 \cdot a_4 \cdot k^2 + a_5 \cdot k. \quad (8)$$

Подставив (8) в (6), получим:

$$t = -\left((a_1 + 2 \cdot a_3 \cdot u_1^* + a_5 \cdot u_2^*) + k \cdot (a_2 + 2 \cdot a_4 \cdot u_2^* + a_5 \cdot u_1^*) \right) / (a_3 + 2 \cdot a_4 \cdot k^2 + a_5 \cdot k).$$

Таким образом, формулы (5) принимают вид:

$$u_1[n] = u_1^* - \left((a_1 + 2 \cdot a_3 \cdot u_1^* + a_5 \cdot u_2^*) + k \cdot (a_2 + 2 \cdot a_4 \cdot u_2^* + a_5 \cdot u_1^*) \right) / (a_3 + 2 \cdot a_4 \cdot k^2 + a_5 \cdot k); \quad (9)$$

$$u_2[n] = u_2^* - k \cdot \left((a_1 + 2 \cdot a_3 \cdot u_1^* + a_5 \cdot u_2^*) + k \cdot (a_2 + 2 \cdot a_4 \cdot u_2^* + a_5 \cdot u_1^*) \right) / (a_3 + 2 \cdot a_4 \cdot k^2 + a_5 \cdot k). \quad (10)$$

Аналіз формул (9) і (10) показує, що $u_1[n]$ і $u_2[n]$ знаходяться в квадратичній залежності від k . Таким чином, не представляється можливим зменшити кількість змінних шляхом отримання лінійних залежностей від k на базі методу Диофанта.

Анализ сокращения объема вычислений на базе решения квадратного уравнения

Допустимо, що в рівнянні (3) $u_2[n]$ має мінімальне значення: $u_2[n] = u_{2,min}$. Тоді (3) приймає вигляд квадратного рівняння з одним невідомим $u_1[n]$:

$$(-y_3 + c_{0[n]}^* + c_{11}^* \cdot y^{(1)}[n+d^*[n]] + c_{12}^* \cdot y^{(1)}[n+d^*[n]-1] + c_{13}^* \cdot u_2,min + c_{14}^* \cdot u_1[n-1] + c_{15}^* \cdot u_2[n-1] + c_{16}^* \cdot u_2^2,min + c_{17}^* \cdot u_1^2[n-1] + c_{18}^* \cdot u_2^2[n-1] + c_{19}^* \cdot u_1[n-1] \cdot u_2,min + c_{20}^* \cdot u_1[n-1] \cdot u_2[n-1]) + (c_{21}^* + c_{22}^* \cdot u_2,min + c_{23}^* \cdot u_2[n-1]) \cdot u_1[n] + c_{24}^* \cdot u_1^2[n] = 0 \quad (11)$$

Данное квадратное уравнение решено для процесса флотации (каналы удельные расходы реагентов собирателя $u_1[n]$ и пенообразователя $u_2[n]$ – зольность флотоконцентрата $y[n]$) [5]; параметры модели (1): период квантования $T_o=1$ с, $d_{min}=390$ с, $d_{max}=460$ с, $y_{min}=3$ %, $y_{max}=13$ %, $u_{1,min}=3000$ г/т, $u_{1,max}=4000$ г/т, $u_{2,min}=7,5$ г/м³, $u_{2,max}=9$ г/м³, $u_{1,min}^*=1$ г/т, $u_{2,min}^*=0,01$ г/м³, $\sigma_{max}=0,5$ %, $k_a=0,5$, $f=0$, $q=1$, $r_1=r_2=0$, $N=1$, $y_3=9,6$ %, $y_{-1}^0=y_0^0=y_{min}=9$ %, $u_{1,s}^0=u_{1,min}=3000$ г/т, $u_{2,s}^0=u_{2,min}=7,5$ г/м³, $N_e^*=180$, $a_{1,1}[n]=-0,9956616120255$, $b_{1,1}[n]=0,004338387974494$, $a_{2,1}[n]=-0,9950124791927$, $b_{2,1}[n]=0,004987520807318$, $a_{1,2}[n]=b_{1,2}[n]=a_{2,2}[n]=b_{2,2}[n]=0$, $d_1[n]=400$ с, $d_2[n]=450$ с, $c_0[n]=0,323$, $c_1[n]=0,0014366485013624$, $c_2[n]=0,855582822$, $c_3[n]=-0,0000001382444$, $c_4[n]=-0,015875644548$, $c_5[n]=-0,000004764213$, $\sigma_n[n]=0,0002$ (%). Расчет коэффициентов уравнения (3) по МНК позволил вычислить коэффициенты в (11): свободный член – (-0,0176100354399973), при $u_1^2[n]$ – (-0,00000000117339969755603), при $u_1[n]$ – (0,00000901508677514656). Количество значащих цифр обусловлено наиболее точным типом *Extended* языка программирования *Delphi*, который использован для исследования МСАУ. Дискриминант квадратного уравнения (11) является положительным числом (251050,323661884). Поэтому корни являются комплексными числами и полученное решение не может быть использовано для аналитического вычисления управляющего воздействия $u_1[n]$, принадлежащего полю рациональных чисел. Таким образом, при синтезе регулятора объекта (1) необходимо и достаточно использование комбинаторного предикторного алгоритма вычисления оптимальных значений управляющих воздействий, который обеспечивает получение наилучшего (в отличие от методов многомерного поиска) решения в реальном масштабе времени в рамках заданного критерия на современных высокопроизводительных промышленных контроллерах [5].

Выводы и перспектива дальнейшего развития

В статье рассмотрены этапы выбора алгоритма оптимального управления нелинейными объектами: использование метода Диофанта, аналитическое вычисление корней уравнения нелинейной модели ТП, комбинаторные предикторные алгоритмы. Для рассматриваемого класса двухканальных, нелинейных, нестационарных, стохастических объектов (1) с большим запаздыванием и диофантовым характером зависимостей на примере ТП УОФ показано: 1) использование метода Диофанта для уменьшения количества переменных является нецелесообразным в связи с квадратичным характером полученных зависимостей (не представляется возможным получить однозначную линейную зависимость двух управляющих воздействий); 2) аналитическое определение корней квадратного уравнения с одним неизвестным (второе фиксируется) приводит к комплексным числам, что вызывает необходимость использования комбинаторных предикторных алгоритмов поиска наилучшего решения. Перспективой дальнейшего развития рассмотренного в статье подхода выбора алгоритма оптимального управления нелинейными объектами представляется совокупное использование методов диофантового анализа [1,2] и теории самоорганизации А.Г.Ивахненко для синтеза оптимальных МСАУ.

Список литературы

1. Башмакова И.Г. Диофант и диофантовы уравнения. – М.: Наука, 1972. – 68с.
2. Спринджук В.Г. Классические диофантовы уравнения от двух неизвестных. – М.: Наука, 1982. – 288с.
3. Романенко В.Д., Игнатенко Б.В. Адаптивное управление технологическими процессами на базе МикроЭВМ: Учеб. пособие. – К.: Выща шк., 1990. – 334с.
4. Микропроцессорное управление многоканальными системами высокой точности / Б.И.Кузнецов, В.Е.Сергеев, В.М.Чернышев. – К.: Тэхника, 1990. – 208с.
5. Зубов Д.А. Автоматичне керування технологічними процесами вуглезбагачувальної фабрики: Монографія. – Луганськ: Вид-во Східноукр. нац. ун-ту ім.В.Даля, 2003. – 172с.
6. Пальчевський Б.О. Дослідження технологічних систем (моделювання, проектування, оптимізація): Навч. посібник. – Львів: Світ, 2001. – 232с.
7. Шуп Т. Решение инженерных задач на ЭВМ: Практическое руководство / Пер. с англ. – М.: Мир, 1982. – 238с.

Зубов Дмитрий Анатолієвич, к.т.н., доц., доцент кафедри комп'ютеризованих систем Восточноукраїнський національний університет імені Володимира Даля (ВНУ ім.В.Даля), Україна, 91034, г.Луганськ, кв.Молодежний, 20а, Телефон (0642) 479261 факс (0642) 413160, dzubovua@mail.ru, dzubov@ccs.snu.edu.ua

UDC 519.682.3

S. Antonakopoulos, I. Diakonikolas

HEURISTIC IMPROVEMENTS TO A LINEAR-TIME APPROXIMATION ALGORITHM FOR THE 2-ECSS PROBLEM

Diakonikolas national technical university of Athens, Greece

Introduction

Communication networks are often modeled by graphs, in which the vertices represent the nodes and the edges represent the feasible communication links. This model allows one to measure the properties and performance of the network by different graph theoretic parameters.

When designing reliable communication networks, the least that we must guarantee is that, after failure of some nodes or links, the surviving network still allows communication between all non-faulty nodes. This implies constraints on the connectivity of the corresponding graph. The k -vertex connectivity (respectively k -edge connectivity) is associated to the capability of a network to resist to the failure of any subset of $k - 1$ nodes (respectively links). A general network design problem which requires the underlying network to be resilient to link failures is known as the Edge Connectivity Survivable Network Design Problem. In fact, 2-edge connectivity is a major feature in today's fast and reliable communication networks, as a single transmission failure could cause intolerable losses.

Most network optimization problems that require finding minimal subgraphs satisfying given connectivity constraints are NP-hard. As a result, it has become imperative to design approximation algorithms for such problems (e.g. [5], [2]). Unfortunately, the 2-edge-connected spanning subgraph problem is proven to be MAX SNP-hard [3], meaning that it cannot be approximated to less than a particular multiplicative constant, unless . On the other hand, several algorithms have been proposed that guarantee satisfactory approximation ratios, starting from $3/2$ [6] and going down to $17/12$ [1], $4/3$ [8], $4/3 - \epsilon$ [7] and finally $5/4$ [4]. However, it is quite unlikely that a lower ratio will be achieved soon, because this would require finding new lower bounds on the optimal solution.

It is worth noting that only the first algorithm, due to Khuller and Vishkin, requires linear time with respect to the number of edges of the input graph. All the rest are significantly more time-consuming, which renders them impractical for use in virtual topologies over large networks, where speed is of the essence. Therefore, in this paper we focus on devising heuristic techniques to improve the aforementioned linear-time algorithm, without increasing its time complexity. We also implement several well-known lower bounds on the optimal solution. Thus, we are able compare the performance of the algorithm with and without heuristics, using random graphs as inputs, and draw useful conclusions.

Heuristics

Our heuristic improvements to Khuller and Vishkin's algorithm [6] involve:

- The choice of the next vertex to be visited by the Depth-First Search routine, according to a primary and a secondary criterion, as implemented in the `selectNextVertex()` function.
- The removal of DFS tree edges which are no longer necessary because of the back edges included in E_H by the algorithm. This optimization takes place after the Depth-First Search and is carried out by the `PostProcessing()` procedure.

The improved algorithm is shown below in pseudocode. In large part, we follow the notation used in [6]. In particular:

- Assume that all vertices are numbered by the order in which they were visited by the DFS, so that the operators $<$ and $>$ make sense when used on vertices.
- Every vertex is initially *unvisited*. When the DFS routine visits it for the first time, it becomes *discovered*. When we finally exit from the vertex, it is marked as *finished*.
- H is the requested spanning subgraph and E_H its edge set.

- $low[v]$ is the vertex with the smallest DFS number that can be reached by following a single back edge originating anywhere in the DFS subtree rooted at v .
- $low_H[v]$ is defined to be the vertex with the smallest DFS number that can be reached by following a single back edge belonging to E_H and originating anywhere in the DFS subtree rooted at v .
- $savior[v]$ is the vertex in the DFS subtree rooted at v which is connected to $low[v]$ by a back edge.

— Algorithm for 2-ECSS —

(* main routine *)

$E_H \leftarrow \square$;

DFSHeur(v_0 , nil);

PostProcessing(v_0);

procedure DFSHeur(v , u); (* u is the parent of v in DFS tree *)

mark v discovered;

$low[v] \leftarrow v$;

$low_H[v] \leftarrow v$;

$savior[v] \leftarrow v$;

while v has unvisited neighbors do begin

$w \leftarrow selectNextVertex(v)$;

$E_H \leftarrow E_H \sqcup \{(v, w)\}$;

DFSHeur(w , v);

$low[v] \leftarrow \min\{low[v], low[w]\}$;

if $low[v]$ changes by the above then $savior[v] \leftarrow savior[w]$;

$low_H[v] \leftarrow \min\{low_H[v], low_H[w]\}$;

end;

for each $w \in Adj[v]$ do

if w is discovered and $w \neq u$ then begin (* (v, w) is a back edge *)

$low[v] \leftarrow \min\{low[v], w\}$;

if $low[v]$ changes by the above then $savior[v] \leftarrow v$;

end;

if $low_H[v] = v$ and $u \neq nil$ then begin

(* edge (u, v) is threatening to be a bridge; add the back edge ($savior[v], low[v]$) to cover it *)

$E_H \leftarrow E_H \sqcup \{(savior[v], low[v])\}$;

$low_H[v] \leftarrow low[v]$;

end;

mark v finished;

end DFSHeur;

function selectNextVertex(v)

for each unvisited $w \in Adj[v]$ do begin

$d_{G-H} \leftarrow$ number of unvisited neighbors of w ;

$x \leftarrow$ lowest discovered neighbor of w (nil if there is none)

(* primary criterion: choose vertex with the fewest unvisited neighbors *)

if $d_{G-H} < bestD$ then begin

$bestD \leftarrow d_{G-H}$; $bestX \leftarrow x$;

$bestVertex \leftarrow w$;

end

(* secondary criterion: choose vertex with the highest lowest discovered neighbor *)

else if $d_{G-H} = bestD$ and ($x = nil$ or $x > bestX$) then begin

$bestX \leftarrow x$;

$bestVertex \leftarrow w$;

end;

end;

return $bestVertex$;

end selectNextVertex;

procedure PostProcessing($root$) (* remove some redundant DFS tree edges *)

for each $v \neq root$ do

redundant \leftarrow false;

$u \leftarrow$ parent of v in the DFS tree;

```

if there exists some  $w$  such that  $u = \text{low}_H[w]$  then begin
if  $(v, \text{low}[v]) \in E_H$  then redundant  $\leftarrow$  true;
if there exist two children  $x, y$  of  $v$  in the DFS tree such that  $\text{low}[x] \neq v$  and
 $\text{low}[y] \neq v$  then redundant  $\leftarrow$  true;
end;
if redundant then  $E_H \leftarrow E_H \setminus \{(u, v)\}$ ; (* edge  $(u, v)$  can safely be deleted *)
end;
end PostProcessing; ■

```

Let y be the vertex with the smallest DFS number that is a neighbor of v . If there exists a descendant w of v such that $y = \text{low}[w]$, then $\text{savior}[v]$ will not be equal to v . This fact is implied in the above code, yet it deserves to be stated explicitly because in some cases it renders the post-processing more effective.

Lower Bounding

The performance of our algorithm is measured against a set of lower bounds on the optimal solution, since it would be overwhelmingly time-consuming to find the optimal solution itself. One obvious lower bound is the number of vertices n . Another is the size of a tree carving of the input graph G , as explained in [6]. This bound is directly related to the number of edges in the solution found by the algorithm, so practically no extra calculations are needed.

We also used a more sophisticated idea, due to Vempala and Vetta [8]. It consists of finding a minimum sized spanning subgraph in which every vertex has degree at least two. The aforementioned problem is called D2. It should be noted that the optimal solution to D2 may not be connected. Given that any two-connected graph must have minimum degree at least 2, it is straightforward that the optimal solution to D2 provides a lower bound on the size of the optimal solution to 2-ECSS.

D2 is solvable in polynomial time by a two-step procedure.

First, find a maximum sized subgraph of G in which every vertex has degree at most two. It is easy to see that the solution to this problem consists of disjoint cycles and paths, which means that the problem asks for a maximum cycle – path cover of the graph. This can be computed efficiently via a simple reduction to maximum bipartite matching, as described below.

Second, add one more edge to each of the end vertices of every path found.

Given a simple undirected graph G with vertex set V and edge set E , construct a bipartite graph H with vertex set $V \cup V$ and edge set $E \cup E$. Find a maximum-cardinality matching M in H . This takes $O(n^3)$ time by using a variant of the Hopcroft – Karp bipartite matching algorithm. It is crucial that we impose the following restriction on the computation of M : for every j, k , at most one of the edges (j, k) and (k, j) is allowed in M .

Let P be the set of edges in G that are in M . The edge set P defines a set of disjoint paths in G . This is true because M is a matching, which means that any vertex j has at most one “incoming” edge (k, j) (corresponding to the at most one edge (k, j)) and at most one “outgoing edge” (j, k) (corresponding to the at most one edge (j, k)). The paths defined by P are all simple and some of them may be closed, i.e. cycles. The time to construct P from M is $O(n)$. Therefore, the time complexity of finding the cover is dominated by the computation of the restricted maximum matching. Under the above constraints, it can be proven that M is of maximum size if and only if P is a maximum cycle – path cover of G .

At this point, the necessity of the imposed restriction should be clear. Both edges (j, k) and (k, j) in the bipartite graph correspond to the same edge (j, k) (of the initial graph). Therefore, the co-existence of such edges in the matching M destroys the maximality of the corresponding cover P .

Conclusions

In this paper, we presented a set of heuristic improvements to a well-known approximation algorithm for the 2-edge-connected spanning subgraph problem, while preserving its linear time complexity. Furthermore, we implemented several lower bounds on the optimal solution, in order to assess the algorithm’s performance. Early tests using random graph inputs of order $n = 10$ to 24 showed that the heuristics produced a solution which was, on average, within 1% of the optimal. On those same inputs, the corresponding error percentage of the original algorithm was ~10%. This improvement is impressive and, since it requires only little more time to be achieved, may be of great practical usefulness. A complete set of computational results will be included in the final version of the paper.

References

1. J. Cheriyan, A. Sebo, Z. Szigeti, “Improving on the 1.5-approximation of a smallest 2-edge connected spanning subgraph”, *SIAM Journal of Discrete Mathematics*, 14:170–180, 2001
2. J. Cheriyan, S. Vempala, A. Vetta, “Approximation algorithms for minimum-cost k -vertex connected subgraphs”, *Proceedings of the 34th ACM Symposium on the Theory of Computing (STOC)*, pp. 306–312, 2002

3. C. G. Fernandes, "A better approximation ratio for the minimum size k -edge-connected spanning subgraph problem", *Journal of Algorithms*, 28:105–124, 1998
4. R. Jothi, B. Raghavachari, S. Varadarajan, "A $5/4$ -approximation algorithm for minimum 2-edge-connectivity", *Proceedings of the 14th annual ACM-SIAM Symposium on Discrete Algorithms*, pp. 725–734, 2003
5. S. Khuller, B. Raghavachari, "Improved approximation algorithms for uniform connectivity problems", *Journal of Algorithms*, 21:433–450, 1996
6. S. Khuller, U. Vishkin, "Biconnectivity approximations and graph carvings", *Journal of the ACM*, 41:214–235, 1994
7. P. Krysta, V. S. Anil Kumar, "Approximation algorithms for minimum size 2-connectivity problems", *Proceedings of the 18th International Symposium on Theoretical Aspects of Computer Science (STACS)*, pp. 431–442, 2001
8. S. Vempala, A. Vetta, "Factor $4/3$ approximations for minimum 2-connected subgraphs", *APPROX* 2000, pp. 262–273, 2000

Spyridon Antonakopoulos, Ilias Diakonikolas National Technical University of Athens, School of Electrical & Computer Engineering, Department of Computer Science 9, Iroon Polytechniou Str., Zografou 15780, Greece, Tel.: +30 210 7721644, E-Mail: {el99612, el99091}@mail.ntua.gr

УДК 681.3.06

В. М. Михалевич

НАВЧАЛЬНО-КОНТРОЛЮЮЧИЙ MAPLE - КОМПЛЕКС З ВИЩОЇ МАТЕМАТИКИ

Вінницький національний технічний університет, Вінниця

Вступ

Останнє десятиріччя характеризується різким розвитком потужних математичних пакетів, що мають приголомшливі можливості в розв'язуванні математичних задач. У всьому світі відбувається осмислення місця і ролі наявних математичних пакетів у викладанні і навчанні дисциплін математичного спрямування. Причому відбувається це на фоні стрімкого розвитку таких пакетів, що супроводжується появою принципово нових властивостей, які розширюють функціональні можливості додатків.

Проте, в даний час широке використання математичних пакетів в освіті стримується, в основному, трьома факторами: відсутністю достатнього числа комп'ютерів і, що ще більш важливо, достатньою кількістю ліцензійних програм з одного боку та відсутністю відповідної підготовки у викладачів – з іншого боку. Що стосується комп'ютерів, то ситуація “змінюється на очах” і вже сьогодні, наприклад, у ВНТУ для переважної більшості студентів ця проблема майже знята. З ліцензійними математичними пакетами ситуація значно складніша. «Купити все» і в «достатній кількості» ніяких грошей не вистачить. Але і цей фактор на сьогодні можна “обійти” використовуючи DEMO версії пакетів. Значно складніше з фактором підготовленості викладачів. Цей фактор посилюється тією обставиною, що, всі найвідоміші, математичні пакети мають стандартні засоби для здобуття тільки остаточної відповіді для типових математичних задач. Для того ж, щоб отримати в середовищі пакета хід розв'язання типової математичної задачі, потрібно мати рівень майстерності володіння пакетом, який набувається роками наполегливої праці. Крім того, математичні пакети не мають стандартних зручних засобів перевірки виконаних завдань. Не кажучи вже про автоматизацію такої перевірки. Отже актуальним є створення навчально-контролюючого комплексу, який є одночасно:

1. Електронним підручником з “живими сторінками”;
2. Програмою генерації умов математичних задач;
3. Тестувальною програмою.

Природно, що такий комплекс може використовуватися як для проведення занять в комп'ютерному класі, так і для дистанційної форми навчання.

В роботах 1, 2 розкривається термін “живі сторінки”. Мова йде про програми, які даватимуть хід розв'язання будь-яких типових задач на запит користувача. Перспективність цієї технології відмічається в 3.

Отже, робота вказаного комплексу базуватиметься на програмах, що здатні відтворити алгоритми символічних перетворень. Природно, що такий комплекс повинен розроблятися в середовищі пакета символічної математики. Як одна з альтернатив для цього може бути вибраний пакет *Maple*. Пошук інших альтернатив можна почати зі знайомства з сайтом 4.

Метою цієї статті є аналіз можливостей додатка *Maple* для створення та використання навчально-контролюючого комплексу.

Для створення електронних підручників *Maple* цілком може замінити текстовий редактор. Цей додаток має зручні засоби структуризації матеріала: розбивка на секції та підсекції з довільним рівнем вкладеності (рис. 1). Гіпертекстові посилання призначені для посилання на інші робочі листи (файли). Документи *Maple* можуть зберігатися в різних форматах: *html*, *rtf*. В 6 запропонована технологія і процедури, що дозволяють підвищити якість *html*-документів, що створюються стандарними засобами пакета *Maple*. Звісно, в *Maple* реалізовано звичайний арсенал будь-якого текстового редактора. Але головним є можливість отримання розв'язку практично будь-якої задачі зі втузівського курсу вищої математики.

В якості ілюстрації пропонується програма розв'язання диференціального рівняння з відокремлюваними змінними (рис. 2).

Наведена програма так само, як і програма, наведена в 1, працює в демо-версії *Maple V R4*. Програма коректно працює практично з усіма диференціальними рівняннями, приведеними в популярних підручниках і збірниках задач з вищої математики для технічних університетів. Для

навчальних цілей це може виявитися цілком достатнім.

Заслужує уваги той факт, що остаточна відповідь, яку дає наведена програма нерідко є більш компактнішою, ніж розв'язок, який дає стандартна команда *dsolve*.

В прикладі, наведеному на рис. 2, остаточна відповідь для одного із розв'язків має вигляд:

The screenshot shows a Maple document interface with a table of contents. The first section is '1 Матриці' (Matrices), which includes sub-sections: '1.1 Поняття матриці' (Concepts of matrices), '1.2 Види матриць' (Types of matrices), '1.3 Дії над матрицями' (Operations on matrices), and '1.4 Матрична форма запису системи лінійних рівнянь' (Matrix form of a system of linear equations). Under '1.3', there is a code block:


```
> resrart:
with(linalg):
Warning, the protected names norm and trace have been redefined and unprotected
```

 followed by a sub-section '1.3.1. Множення матриці на число або числа на матрицю - результатом є матриця, елементами якої є добутки елементів даної матриці на це число.' (Multiplication of a matrix by a scalar or scalars by a matrix - the result is a matrix, the elements of which are the products of the elements of the given matrix by this scalar). The second main section is '2 Визначники' (Determinants), with sub-sections: '2.1 Поняття визначника матриці другого порядку' (Concepts of the determinant of a second-order matrix), '2.2 Властивості визначника матриці другого порядку' (Properties of the determinant of a second-order matrix), '2.3 Поняття та властивості визначника матриці третього порядку' (Concepts and properties of the determinant of a third-order matrix), '2.4 Перевірка властивостей визначника' (Checking the properties of the determinant), '2.5 Обчислення визначника за теоремою розкладання' (Calculation of the determinant by the expansion theorem), and '2.6 Теорема про анулювання' (Theorem on annihilation).

Рис. 1. Структуризація Maple – документів 5

$$2\sqrt{y} = \frac{2}{3}x^{\left(\frac{3}{2}\right)}\sqrt{k} + C$$

Аналогічна відповідь, здобута за допомогою стандартної команди *dsolve* повної версії *Maple V R4* має вигляд:

$$\frac{2}{3} \frac{(k y(x) x)^{(3/2)}}{k y(x)^{(3/2)}} + 2\sqrt{y(x)} - \frac{\sqrt{k y(x) x} \ln(y(x)) x}{\sqrt{y(x)} \sqrt{k x}} + \frac{(k y(x) x)^{(3/2)} \ln(y(x))}{y(x)^{(3/2)} (k x)^{(3/2)} k} = -C1$$

Відповідь, яку дає Maple 7, має вигляд

$$y(x) = \frac{1}{9} \frac{(2 k - C1 x^2 + 2 \sqrt{-C1 k x}) x - C1 x^3 k + 1}{-C1}$$

Maple 9:

$$y(x) = - \frac{(-2 _CI x^2 k + 2 \sqrt{_CI k x}) x + _CI k x^3 - 1}{9 _CI}$$

Важливим є не просто компактність остаточної відповіді, а збіг її вигляду з тим виразом, який отримує студент, дотримуючись типового алгоритму.

```

Maple V Release 4 - [SeparEq_R4_Ukr.mws - Demo Edition]
File Edit View Insert Format Options Window Help
[ Розв'язати диференціальне рівняння з відокремленими змінними:
DEMO > restart:
diff(y(x),x)^2=k*y(x)*x: # Вихідне рівняння
odel:="":odel;

      (∂/∂x y(x))^2 = k y(x) x

[ Розв'язання:
[ Розв'яжемо рівняння відносно похідної:
DEMO > fxy:=solve(odel,diff(y(x),x)): #Розв'язанн_р_івн_нн_
map(x -> x, [fxy]):sxn:=nops("):
if sxn>1 then
    print(`Дане р_івн_нн_ розпадаєтьс_ на декілька р_івн_нь:`);
    for kj from 1 to sxn do
        print(dy/dx=fxy[kj])
    od;
    fxy:=fxy[1]:
    print(`Далі буде розгл_нуто таке р_івн_нн_`);
fi:
if (has type(fxy, integer^anything) or hasfun(fxy,exp,[x,y])) then
    if type(fxy,`*`) then xyl:=1:
        for k from 1 to nops(fxy) do
            xyl:=xyl*expand(op(k,fxy))
        od;fxy:=xyl
    elif type(fxy,``) or type(fxy,function) then fxy:=expand(fxy)
fi:fi:

```

Рис. 2. Програма для отримання ходу розв'язання диференціального рівняння з відокремленими змінними

В режимі навчання програми, що подібні наведеній, дозволяють студенту відворити весь хід розв'язання будь-яких типових задач вищої математики за стандартними алгоритмами. Умови задач студент може отримувати за допомогою блоку генерації задач, або з інших доступних джерел. Унікальною особливістю комплексу є те, що студент має змогу познайомитись не тільки з довільною кількістю типових задач, а й змінювати умови задачі під свої особисті питання. В результаті з'являється новий ефект в навчанні: студент сам собі задає питання і сам отримує відповідь на нього. Природно, що у кожного студента виникають «свої питання», чого не в змозі врахувати будь-який статичний підручник.

При роботі комплексу в режимі контролю повинен бути відключений доступ студентів до всіх програм, що дають хід розв'язання типових задач. Але ці програми використовуються для перевірки відповідей, які пропонуються студентом. Причому потужні засоби *Maple* з успіхом виконують операції ідентифікації різних форм відповідей. При цьому перевірка може здійснюватися не тільки за остаточною відповіддю, а й за ключовими проміжними результатами. Для розглянутого прикладу такими результатами можуть бути класифікація рівняння та диференціальне рівняння з відокремленими змінними, кожний із знайдених інтегралів. Такий контроль буде повнішим, а значить і об'єктивнішим. Студент, який зробив помилку, при знаходженні одного із інтегралів, або в несуттєвих перетвореннях остаточної виразу, і, таким чином, отримав невірну остаточною відповідь, може отримати високу позитивну оцінку, якщо всі проміжні результати отримано правильними.

Безумовно, одним із серйозних недоліків аналізованого проекту навчально-контрольного комплексу, що аналізується, є надзвичайна трудомісткість його створення. Але для використання комплексу зовсім не обов'язково чекати його повної готовності. Автор уже декілька років з успіхом використовує створені елементи комплексу (процедури) для проведення лабораторних та практичних робіт, організації самостійної роботи студентів, генерування типових розрахункових завдань.

Оскільки ліцензійна версія пакета *Maple*, навіть для навчальних закладів, дуже дорога, для масового використання підготовлених програм пропонується *DEMO* версія *Maple*, інсталяційний пакет, якої займає 2 Мб.

```

Maple V Release 4 - [SeparEq_R4_Ukr.mws - Demo Edition]
File Edit View Insert Format Options Window Help
dy/dx=fxy; #Виведенн_ результату на дисплей
fxy:=factor(fxy):
DEMO >
Дане рівн_нн_ розпадаєтьс_ на декілька рівн_нь:

$$\frac{dy}{dx} = \sqrt{k y(x) x}$$


$$\frac{dy}{dx} = -\sqrt{k y(x) x}$$

Далі буде розгл_ нуто таке рівн_нн_

$$\frac{dy}{dx} = \sqrt{k y(x) x}$$

[ або
DEMO > fxy:=subs(y(x)=y, fxy):
dy/dx=fxy;

$$\frac{dy}{dx} = \sqrt{k y x}$$

[ Розділимо змінні
DEMO > if not has(fxy,y) then f2y:=1;flx:=fxy
elif not has(fxy,x) then f2y:=fxy;flx:=1
else
if hastype( fxy, radical ) and not type(fxy,`+`) then
if remove(hastype,fxy,radical)=fxy then if type(op(1,fxy),`*`) then
mul((op(i,op(1,fxy)))^(op(2, fxy))), i=1..nops(op(1, fxy)));fxy:=" fi;
else

```

Рис. 2. (продовження)

```

Maple V Release 4 - [SeparEq_R4_Ukr.mws - Demo Edition]
File Edit View Insert Format Options Window Help
ST:=select(hastype, fxy, radical);
if type(op(1,ST),`*`) then
remove(hastype, fxy, radical)*mul((op(i,op(1,ST)))^(op(2,ST))),
i=1..nops(op(1,ST));fxy:=" fi;
fi;fi;
f2y:=select(has, fxy, y):#Виділенн_ із правої частини функцій, _кі
утримують `y`
flx:=remove(has, fxy, y):#Вилученн_ із правої частини функцій, _кі
утримують `y`
f2y:=`if`(f2y=NULL,1,f2y):#_кщо в правій частині відсутній `y`, то
f2y=1
fi:
#Блок перевірки на відповідність типу ДР з відокремлюваними змінними
#*****
*
if type(fxy,`+`) and has(fxy,x)*has(fxy,y)=true^2 then
print(`Невідповідність типу.`);
elif type(f2y,string)*type(flx,string)=string^2 then
print(`Невідповідність типу диференціального
рівн_нн_`);
elif has(f2y,x)*has(flx,y)=false^2
then
(1/f2y)*dy=flx*dx;#Виведенн_ ДР з відокремлюваними змінними
на дисплей
else
print(`Невідповідність типу диференціального рівн_нн_`);

```

Рис. 2. (продовження)

The screenshot shows the Maple V Release 4 interface. At the top, the title bar reads "Maple V Release 4 - [SeparEq_R4_Ukr.mws - Demo Edition]". The menu bar includes File, Edit, View, Insert, Format, Options, Window, and Help. The main workspace contains the following text and equations:

```
f1;
```

$$\frac{dy}{\sqrt{y}} = \sqrt{k} \sqrt{x} dx$$

Дістали диференціальне рівняння з відокремленими змінними. Обидві частини цього рівняння можна проінтегрувати :

```
DEMO > My_C := C;
        Int (1/f2y, y) = Int (f1x, x) + My_C;
        y := 'y':
```

$$\int \frac{1}{\sqrt{y}} dy = \int \sqrt{k} \sqrt{x} dx + C$$

Загальний розв'язок диференціального рівняння:

```
DEMO > int (eval (1/f2y), y) = int (f1x, x) + My_C;
```

$$2\sqrt{y} = \frac{2}{3} x^{3/2} \sqrt{k} + C$$

Рис. 2. (продовження)

Автором розроблена технологія, яка дозволяє "обійти" цілий ряд суттєвих обмежень DEMO версії. Важливо, що програми, подібні, до наведеної, не тільки розширюють функціональність ліцензійної версії, а й в цілому ряді випадків значно послабляють обмеження DEMO версії. Наприклад, в DEMO версії заблокована стандартна команда *dsolve* для здобуття символьного розв'язку диференціального рівняння.

Висновки

1. Пакет символьної математики є зручним середовищем для створення навчально-контролюючого
2. Створенням нових елементів комплексу та вдосконаленням існуючих повинно відбуватися паралельно з використанням наявних елементів.
3. Массове використання студентами указаних елементів на сьогодні можливе з застосуванням DEMO версії Maple.
4. По наближеним оцінкам, на створення першої версії повнофункціонального комплексу потрібно декілька років. Є підстави сподіватись, що на той час і цінова політика на математичні пакети буде менш жорсткою, зокрема, для вітчизняних університетів.

Список літератури

1. Михалевиц В.М. Реалізація технології "живих сторінок" в Maple, MathCad, Excel // Вісник ВПІ. – 2004. – № 3. – С. 90-95.
2. Mikhalevich V. "Development of Electronic Courses in Maple Environment on a Process Engineering of a "Live Pages". // "Інтернет – Освіта - Наука - 2004", четверта міжнародна конференція ІОН – 2004, 28 вересня – 16 жовтня, 2004 р. Збірник матеріалів конференції. Том 1. – Вінниця: Універсум-Вінниця, 2004. – С.31-34.
3. Мокін Б.І., Мокін В.Б., Мокіна О.О. Концепція створення дистанційної форми навчання в магістратурі Вінницького національного технічного університету // Вісник ВПІ. – 2004. – № 1. – С. 108-115.
4. <http://www.exponenta.ru/>
5. Михалевиц В. М. Maple. Комп'ютерна підтримка курсу вищої математики в технічному вузі. Частина I. Лінійна й векторна алгебра. Аналітична геометрія. Навчальний посібник. - Вінниця: ВНТУ, 2004. - 111 с.
6. Аладьев В.З. Эффективная работа в Maple 6/7 – М.: Лаборатория Базовых Знаний, 2002. – 336 с.
7. Мокін Б.І., Мокін В.Б., Мокіна О.О. Технологія швидкої підготовки лекцій для дистанційної форми навчання // Вісник ВПІ. – 2004. – № 2. – С. 89-94.

Михалевиц Володимир Маркусович, доктор технічних наук, професор зав. кафедри прикладної математики ВНТУ. Тел. (8-0432) 44-05-94, E-Mail: mikhal@svitonline.com

УДК 681.3.06

А. И. Толкачёв

ПРИМЕНЕНИЕ ПРОГРАММНОГО КОМПЛЕКСА РАЗРАБОТКИ КОМПИЛЯТОРОВ ДЛЯ ПАРАЛЛЕЛЬНЫХ ВЫЧИСЛИТЕЛЬНЫХ СИСТЕМ В ПРОЦЕССЕ ОБУЧЕНИЯ СПЕЦИАЛИСТОВ

Гомельский государственный университет имени Франциска Скорины, Республика Беларусь

Вступление

В последнее время наблюдается большой интерес разработчиков программного и аппаратного обеспечения вычислительных систем к использованию параллельной обработки.

Это обусловлено, во-первых, быстрым развитием сфер применения встроенных систем и специализированных цифровых устройств. К таким системам постоянно предъявляются новые требования производительности и возможности решения различных задач, таких как обработка звука, изображений, видео, шифрование и сжатие данных и т.д.

Во-вторых, использование параллельной обработки является единственным способом решения некоторых задач, требующих огромных объемов вычислений, например, создание качественного видеоизображения, прогнозирование погоды, моделирование физических процессов.

Разработанный нами программный комплекс проектирования компиляторов языков высокого уровня для параллельных вычислительных систем может использоваться для обучения специалистов в области параллельного программирования. В основе программного комплекса лежит алгоритм автоматического выявления параллелизма [1, 2], реализованный в виде универсального модуля. Настройка на целевую платформу осуществляется реализацией *back-end* компилятора, принимающего на входе максимально распараллеленное промежуточное представление.

Проблемы проектирования программного обеспечения для параллельных архитектур

Основная проблемы, встающие при разработке программ для параллельных архитектур, заключаются, во-первых, в эффективности использования имеющихся ресурсов и, во-вторых, в сложности написания программ, максимально эффективно использующих возможности целевой архитектуры.

Системы с распределенными ресурсами включают в себя множество различных вариантов конфигураций вычислительных систем, и для каждой из них существуют различные критерии эффективности. Если для традиционных последовательно работающих процессоров критерии оптимизации примерно одинаковы, то для параллельных систем они могут существенно различаться. Например, в многопроцессорном компьютере может потребоваться оптимизация обращений к общей памяти, а в вычислительной системе, состоящей из множества компьютеров, работающих в локальной сети, узким местом может быть скорость передачи данных между узлами.

Перечисленные проблемы приводят к тому, что для разработки программ для параллельных вычислительных систем используются различные методы и инструменты.

Существующие средства для программирования вычислительных систем с распределенными ресурсами

Существует множество систем для создания программного обеспечения для систем с распределенными ресурсами. Можно выделить несколько классов средств разработки. Самыми многочисленными являются системы, представляющие собой библиотеки для некоторого языка программирования высокого уровня. Они, как правило, используются при программировании для систем, состоящих из множества параллельно работающих компьютеров, и содержат функции для синхронизации процессов и передачи данных между ними. Вся работа по распараллеливанию алгоритма возлагается на программиста. Самой распространенной системой такого типа является *MPI* [6] (*Message*

Passing Interface, інтерфейс для передачі повідомлень).

Друга технологія – *OpenMP* [6], може використовуватися в багатопроцесорних системах з загальною пам'яттю. Ідея заключається в тому, що певні фрагменти програми можуть бути оголошені паралельними, і тільки вони можуть виконуватися одночасно. Завжди існує основний процес, який створює паралельні і очікує закінчення їх роботи. Як і в разі з *MPI*, розпаралелювання здійснюється програмістом.

З систем, використовуючих розширення мов програмування, розглянемо *mpC* [6]. Програма на *mpC* являє собою опис множини процесів, взаємодіючих з допомогою неявної передачі повідомлень. Розподіл процесів між фізичними процесорами відбувається автоматично в процесі виконання програми.

Засоби для програмування суперскалярних і *VLIW* процесорів являють собою оптимізуючі компілятори або транслятори асемблера, розроблювані або налаштовані спеціально для кожної архітектури. Можливості паралельної обробки *VLIW* процесорів мають суттєві обмеження, що дозволяють оптимізувати програму тільки на рівні інструкцій.

Актуальні проблеми, мета дослідження

Актуальною є проблема автоматизації проектування програмного забезпечення для паралельних архітектур. Метою дослідження є створення універсальних методів компіляції мов високого рівня для паралельних висувальних систем, а також реалізація програмного комплексу для автоматизованого проектування компіляторів.

В ході дослідження вирішувалися наступні проблеми:

- автоматизація розробки синтаксических аналізаторів входних мов високого рівня;
- створення формату проміжного представлення програми, яке може містити інформацію про паралелізм, використовуватися для застосування різних алгоритмів оптимізації, а також є універсальним по відношенню до входної мови і цільової платформи;
- розробка алгоритмів автоматичного виявлення паралелізму, що визначають можливості паралельного виконання як на рівні окремих інструкцій, так і цілих блоків вихідної програми;
- алгоритми оптимізації, що дозволяють розробляти ефективні компілятори для паралельних висувальних систем з різними обмеженнями і вузькими місцями;
- інтеграція отримуваних компіляторів з універсальною середовищем для розробки програмного і апаратного забезпечення вбудованих систем.

Пропонується технологія проектування компіляторів для систем з розподіленими ресурсами

Основна ідея проектування компіляторів для висувальних систем з розподіленими ресурсами заключається в створенні універсальної незалежної від вихідної мови і цільової платформи частини компілятора. Цей блок компілятора (розпаралелюючий компілятор) здійснює аналіз програми і виявляє всі можливості паралельного виконання як на рівні простіших операторів (арифметических, логіческих), так і цілих блоків програми (циклів, функцій і др.). Вхідними даними для роботи розпаралелюючого компілятора є проміжне представлення вихідної програми, отримане в результаті роботи *front-end* компілятора. Отримане на виході проміжне представлення виконуваного коду містить інформацію про всі знайдені можливості паралельного виконання. Генерацію виконуваного коду для заданої цільової платформи виконує *back-end* компілятор, що отримує розпаралелене представлення програми.

Пропонується схема універсального компілятора зображена на малюнку 1.

Розроблений метод проектування компіляторів для висувальних систем з розподіленими ресурсами покриває весь цикл розробки компілятора, починаючи з синтаксического аналізу вихідної програми і закінчуючи генерацією виконуваного коду.

Основними етапами проектування компілятора є наступні:

Розробка компілятора вихідного ЯВУ в послідовне проміжне представлення. Для реалізації синтаксического аналізатора вихідної мови використовується програмний комплекс *Unisan* [3].

Реалізація алгоритмів розпаралелювання. Результатом їх роботи є проміжне представлення програми, що містить інформацію про всі знайдені можливості паралельного виконання.

Створення платформено – незалежного оптимізатора проміжного представлення, що містить інформацію про паралелізм.

Реалізація оптимізатора, що враховує особливості цільової архітектури.

Разработка генератора исполняемого кода для целевой платформы.

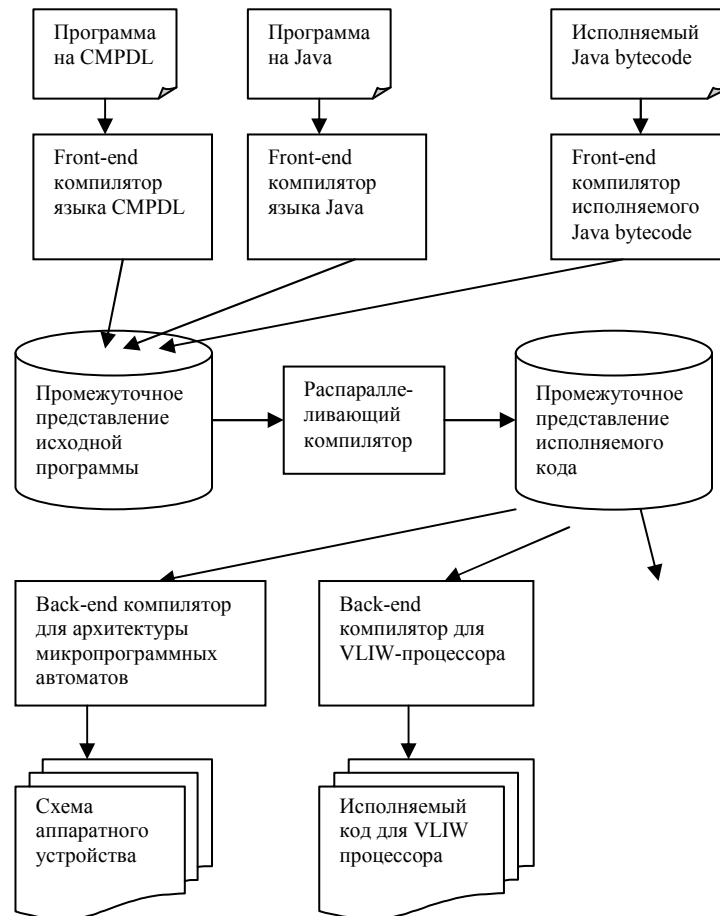


Рис. 1. Схема настраиваемого компилятора языков высокого уровня для параллельных вычислительных систем

Новизна метода проектирования распараллеливающих компиляторов заключается в следующем:

Алгоритм факторизации синтаксических диаграмм в универсальном синтаксическом анализаторе *Unisan* позволяет значительно расширить допустимый класс грамматик, и, соответственно, уменьшить трудоемкость описания синтаксиса входного языка.

Универсальные алгоритмы распараллеливания позволяют выявлять возможности параллельного исполнения как на уровне отдельных инструкций, так и целых блоков исходной программы.

Формат промежуточного представления программы может содержать информацию о параллелизме, использоваться для применения различных алгоритмов оптимизации, а так же является универсальным по отношению к входному языку и целевой платформе.

Получаемая в результате работы распараллеливающего компилятора информация может быть использована для генерации исполняемого кода для широкого класса архитектур с распределенными ресурсами.

Алгоритмы оптимизации позволяют разрабатывать оптимизирующие компиляторы для параллельных вычислительных систем с различными ограничениями и узкими местами.

Интеграция с универсальной отладочной средой для разработки программного и аппаратного обеспечения встроенных систем снижает трудоемкость и стоимость разработки программного или аппаратного обеспечения, использующего параллельную обработку.

Применение разработанных средств для обучения специалистов

Благодаря возможности повторного использования универсальных блоков компилятора значительно снижены необходимые объем работы и трудоемкость реализации оптимизирующих компиляторов для вычислительных систем с распределенными ресурсами. Это позволяет применять программный комплекс для обучения специалистов в области проектирования компиляторов для параллельных вычислительных систем.

В качестве апробации разработанных средств в Гомельском государственном университете реализован компилятор языка *CMPDL* [4] (подмножество языка Си) для архитектуры микропрограммных автоматов [4]. Результатом работы компилятора является синтезируемое описание аппаратной схемы, которая реализует описанный алгоритм с использованием параллельной обработки. Интеграция компилятора с системами высокоуровневого проектирования аппаратного и программного обеспечения встроенных систем *Winter* и *IEESD* [5] позволяет разработчикам моделировать и отлаживать работу устройства с использованием программных моделей процессора микропрограммных автоматов и устройств окружения, а затем получать синтезируемое описание устройства на языке *Vhdl* [7]. Синтезируемое описание может быть аппаратно реализовано в ПЛИС или СБИС.

Программный комплекс успешно используется в процессе обучения студентов специальности “Прикладная математика” по курсу “Проектирование цифровых устройств”.

Выводы, перспективы дальнейшего развития

Разработанная технология проектирования компиляторов для систем с распределенными ресурсами может применяться для создания компиляторов практически любых языков программирования. В качестве целевой платформы возможно использование широкого класса вычислительных систем с распределенными ресурсами, в частности, VLIW процессоров, многопроцессорных систем, генератора аппаратных схем и др.

Практическая значимость заключается в значительном снижении трудоемкости реализации оптимизирующих компиляторов для параллельных вычислительных систем по сравнению с другими средствами проектирования компиляторов. Разработанный программный комплекс может использоваться для обучения специалистов.

Ограничение технологии состоит в отсутствии встроенных средств коммуникации при использовании многопроцессорных целевых архитектур или вычислительных систем, состоящих из множества компьютеров. При создании компиляторов для таких архитектур необходимо дополнительно реализовывать коммуникационные средства для взаимодействия компонент вычислительной системы. Разработка и программная реализация универсальных средств коммуникации является одной из перспектив развития.

Другим направлением развития является создание универсальных средств, автоматизирующих разработку эффективных генераторов исполняемого кода для целевых архитектур со специфическими инструкциями, не имеющими аналогов в промежуточном представлении.

Список литературы

1. Tolkachiov A., Dolinsky M. Hardware Implementation of Complex Data Processing Algorithms // Proceedings of the Work in Progress Session held in Connection with the 29th EUROMICRO Conference EUROMICRO 2003 and the EUROMICRO Symposium on Digital System Design DSD 2003, Belek (Turkey), September 2003, Published by Institute of System Science Johannes Kepler Universitat Linz, Austria, ISBN 3-902457-21-X.
2. Толкачѳв А.И. Комплекс для проектирования аппаратных решений, эффективно реализующих сложные алгоритмы обработки данных // Известия Гомельского государственного университета имени Ф.Скорины. – 2003. – №3(18), – с.153-157.
3. Толкачѳв А.И. Использование программного комплекса автоматизированной разработки трансляторов в учебном процессе // Интернет-Образование-Наука-2002: Материалы 3-й международной конференции – Винница: Універсум-Вінниця – с. 100-102.
4. Долинский М., Коршунов И., Толкачев А., Ермолаев И., Литвинов В. Технология разработки алгоритмически сложных цифровых систем с помощью автоматического синтеза микропрограммных автоматов // Компоненты и технологии. – 2003. – №8. – с.110-114.
5. Долинский М.С., Ермолаев И.Ю., Гончаренко И.И., Толкачѳв А.И. Winter - среда отладки программного обеспечения мультипроцессорных систем // Компоненты и технологии. – 2003. №2, – с.63-69.
6. В.В. Воеводин, Вл.В.Воеводин Параллельные вычисления. – СПб.: БВХ-Петербург, 2002.
7. Chang K.C. Digital Design and Modeling with VHDL and Synthesis // IEEE Computer Society Press, Los Alamitos, California. - Los Alamitos, California. – 1997.

Алексей Иванович Толкачѳв, Гомельский Государственный университет имени Франциска Скорины, ул. Советская, 104, Гомель, 246000, Республика Беларусь, Tel. +(375) 232 578256. E-mail: tolkachyov@gsu.unibel.by

УДК 371.3

О. І. Гороховський, Т. І. Трояновська

ДОСЛІДЖЕННЯ СТРУКТУРИ ДИСТАНЦІЙНИХ КУРСІВ

Вінницький національний технічний університет, Вінниця

Вступ

Принциповим в ідеї дистанційної освіти (ДО) є перехід від прийнятого в традиційних видах навчання (як очному, так і заочному) "руху студентів за знаннями" до "руху знань до студентів". Використання методів ДО дозволяє одержувати якісне навчання у віддалених районах, реально учитися без відриву від основної роботи, навчати осіб з обмеженими фізичними можливостями, значно знизити транспортні витрати для тих хто отримує освіту і т.д. Використання сучасних засобів електронних комунікацій дозволяє забезпечити з помірними фінансовими витратами, контакт між студентом і викладачем, віддаленими один від одного на сотні або тисячі кілометрів.

Електронний підручник є ключовим моментом заочно-дистанційної форми навчання. Незважаючи на те, що термін "Електронний комп'ютерний підручник" (ЕКП) здобуває усе більше поширення, різні автори вкладають у нього істотно різний зміст. Єдине загальноприйняте визначення відсутнє, однак ясно, що його не можна зводити тільки до одного з численних видів навчальних програм. Досить розповсюдженим є погляд на ЕКП, як на програмно-методичний комплекс, що дозволяє самостійно освоїти навчальний курс або його великий розділ, і часто такий, що об'єднує у собі властивості звичайного підручника, довідника, задачника і лабораторного практикуму. Використання його не альтернатива, а доповнення до традиційних форм навчання, і не замінює роботу студента з книгами, конспектами, збірниками задач і вправ і т.п. Цей "електронний лектор" покликаний не тільки зберегти всі переваги книги або навчального посібника, але і повною мірою використовувати сучасні інформаційні технології, мультимедійні можливості, надані комп'ютером.

Однак на сьогодні ще немає чіткого стандарту на склад і структуру як ЕКП так і дистанційного курсу в цілому, що значною мірою стримує процес розробки дистанційних курсів і впровадження ДО в цілому. У статті проаналізовано різні аспекти створення дистанційного курсу і запропоновано один з можливих варіантів його структури.

Технологічний аспект розробки і підтримки системи дистанційної освіти (СДО)

Створення повноцінного середовища розробки і підтримки СДО вступило в стадію реалізації. У якості базової використовується технологія "World Wide Web" - існуюча більш як 10 років технологія доставки інформації користувачеві мережі Інтернет. Система заснована на парадигмі гіпертекст/гіпермедиа. Базовий протокол - HTTP, основна мова розмічування гіпертекстових документів - HTML (<http://www.w3.org>). Компанія *Javasoft* визначає дану технологію як найбільш сучасний засіб для розробки прикладних програм, у тому числі розподілених, для Інтернету [<http://www.javasoft.com/>].

Організаційні аспекти створення СДО

Перед кожним викладачем рано чи пізно виникне задача завоювання, збереження і розширення власної аудиторії слухачів. Цього не можна домогтися тільки впровадженням новітніх комунікаційних технологій у процес навчання. Необхідно скористатися всіма доступними інформаційними засобами для організації ефективних альтернативних форм навчання.

У дистанційному навчанні форма представлення навчальних матеріалів здобуває особливе значення. По можливості, в навчальних матеріалах заздалегідь повинні бути закладені відповіді на всі питання, що можуть виникнути у студента з приводу методики вивчення курсу. Для збереження й обміну файлами колективних проєктів, необхідно передавати на Web-сервер велику кількість файлів, тому студентам необхідно надати доступ до файлової системи на сервері. Звичайно, користувач не повинен мати доступ до всіх каталогів на сервері. Досить однієї папки для збереження всіх файлів користувача. При доступі до файлової системи необхідно організувати обмін інформацією на рівні архівів. Це дозволить знизити мережний трафік при передачі великих файлів і полегшить процедуру одночасної передачі великої кількості файлів. При цьому бажано обмежити розмір дискового простору, займаного користувачем.

При організації дистанційного навчання гостро стоїть питання виконання студентами практичних робіт на комп'ютері. Програмне забезпечення (ПЗ), що найчастіше використовується у навчальному процесі, не може бути перенесене на домашній комп'ютер студента. Крім того, при передачі такого ПЗ,

можуть бути порушені ліцензійні угоди.

У зв'язку з цим, найефективнішою є можливість організації термінального доступу до серверів навчального закладу, з яких студент одержує на свій комп'ютер лише результати, у той час як самі програми виконуються на сервері. При дистанційному освоєнні навчально-кваліфікаційних програм інженерної підготовки (лабораторні роботи, практичні завдання) у деяких випадках ефективнішим буде навіть не використання спеціального ПЗ, а застосування відео-інтернетного комплексу (ВІК) [1].

Проектування СДО

Для того, щоб курс був зручним у використанні, необхідно продумати його структуру з погляду його окремих складових:

1. Можливі дії учасників ДО:

Абітурієнт може безкоштовно ознайомитися із демо-версіями лекцій, одержати інформацію про реєстрацію і про правила роботи з курсом, сформулювати власну думку про курс і написати відгук. Основні можливості абітурієнта зображені на рис. 1.

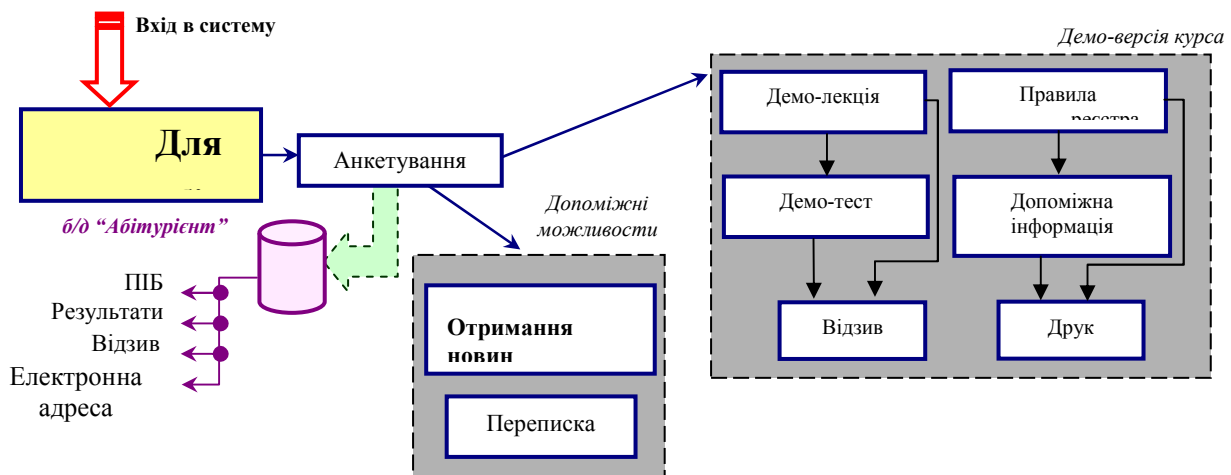


Рис. 1. Вхід в систему для абітурієнта

Студент вивчає обраний курс, проходить тестування або атестацію іншого роду. Основні можливості студента подані на рис.2.

Викладач консультує студентів, стежить за проходженням контрольних заходів, їхньою результативністю, видає додаткові завдання, допуски; за необхідністю, проводить семінари і читає лекції, а також виставляє оцінки. Основні задачі і функції викладача подані на рис.3.

Адміністратор має необмежений доступ до всіх ресурсів, забезпечує безпеку системи, і несе відповідальність за систему в цілому. Можливості адміністратора подані на рис. 4.

2. Пошук інформації. В дистанційний курс необхідно включити систему пошуку, для того, щоб студент міг скористатися перевагами мережі і швидко знайти необхідні матеріали. Незважаючи на те, що саме ДО розрахована на досвідчених користувачів, не зашкодить зробити посилання: Як користуватися пошуком?

3. Служба допомоги (HELP), дає конкретні відповіді на питання про зміст курсів, перелік предметів, вимоги, довідкову інформацію, термінологію, та ін.

4. Навігація. Навігаційна система підручника повинна будуватися за блоковим принципом з ієрархічною перехресною структурою посилань усередині кожного блоку. За необхідності, можуть бути реалізовані і переходи між різними блоками. Перехресна навігація між різними сторінками підручника повинна реалізовуватися тільки в неявному виді {вигляді}. На кожній сторінці підручника в обов'язковому порядку повинні бути присутні дві навігаційні панелі: на початку і в кінці сторінки. Таке розташування кнопок найприродніше і забезпечує максимальні зручності в роботі [2].

5. Кнопки повинні бути не занадто маленькими і не дуже великими через те, що маленькі будуть важкими для сприйняття (особливо, якщо на кнопки є ще і текст!), а великі будуть "з'їдати" і без того потрібний екранний простір. Є такі користувачі, що відключають графіку, якщо працюють в Інтернеті, тому треба використовувати завжди альтернативні підписи для графічних зображень.

6. Способи представлення тексту. На процес сприйняття (виходить, і розуміння) тексту, розташованого на екрані дисплея впливає цілий набір факторів: ширина текстової зони, спосіб вирівнювання тексту, його розташування на екранній сторінці, накреслення, стиль і розмір шрифту.

7. Текст. Сприйняття тексту йде значно ефективніше, якщо око може відразу охопити не окремі слова або обривки фрази, а цілий рядок, закінчений фрагмент, в ідеалі – абзац.

8. Зворотний зв'язок. Для того, щоб студент міг контактувати з викладачем, необхідно передбачити спілкування. Наприклад, електронну пошту, чат, форум.

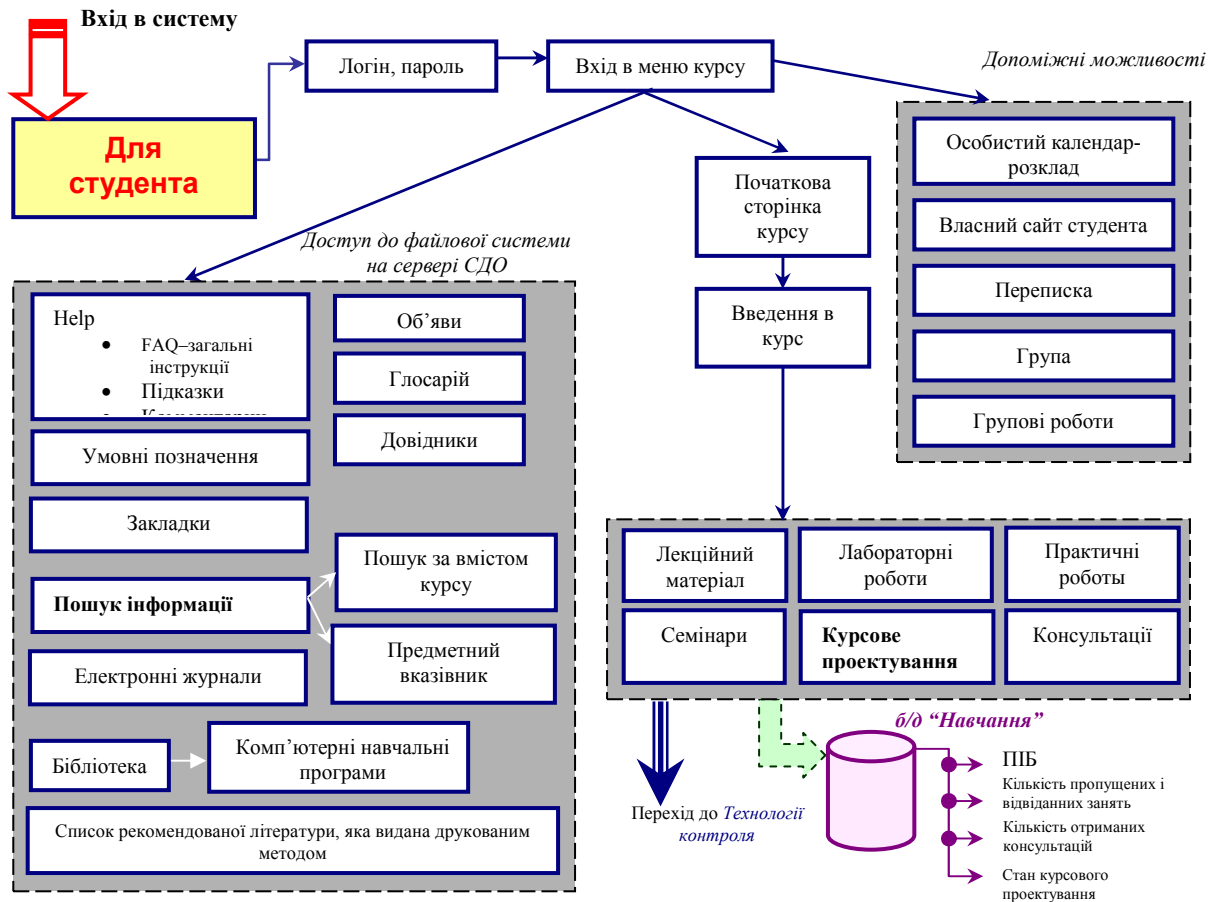


Рис. 2. Вхід в систему для студента

Методичні аспекти створення СДО

Однією з найважливіших вимог, що висувається до інструментів для створення Web-курсів, є вимога до дружності інтерфейсу. Типовими недоліками наявних систем проектування Web-курсів є нестандартне розташування елементів управління і велика глибина і складність ієрархії меню. Представити студентів курси в системі із продуманим інтерфейсом так само важливо, як створювати Web-сайт із вдалим дизайном.

Весь комплекс програмної і методичної документації ДО поділяється на дві частини. Перша частина – це документи, що розробляються адміністраторами і методистами ДО (навчальний план, загальні інструкції (FAQ), семестровий план - графік). Друга частина розробляється викладачами і тьюторами (курсний опис, дистанційний курс, електронний підручник). Всі ці три види матеріалів вишиковуються по такому принципові: кожний наступний містить у собі попередній (крім повторів).

В ієрархії методичних матеріалів дистанційного навчання дистанційний курс за обсягом і за ступенем відображення навчального матеріалу займає проміжне положення між курсовим описом і електронним підручником. Дистанційний курс – це аналог навчально-методичного посібника для студентів, що навчаються за традиційною заочною і екстерною формами. Використовуючи іншу аналогію, дистанційний курс можна назвати путівником по навчальній дисципліні з усіма властивими йому атрибутами.

Фактично дистанційний курс – це розроблена з визначеним ступенем детальності покрокова інструкція з освоєння курсу, тобто з досягнення цілей, сформульованих в його описі.

Як правило, дистанційні курси розробляються за допомогою різних програмних продуктів, що дозволяють представляти його зміст у форматах Інтернету (HTML). Це можуть бути як найпростіші розробники веб-сайтів, вбудовані в текстові редактори, так і такі програми, як MS FrontPage, MS Publisher і ін. Курси можуть розроблятися і за допомогою спеціальних програм-оболонки, призначених

для дистанційного навчання (*WebCT, Lotus Learning Space, BlackBoard, Доцент, Прометей* і т.д.). Відмінність лише в тому, що в спеціальних програмах відразу задається шаблон структури курсу.

Сам процес розробки курсу чітко поділяється на дві частини:

- Розробка методичного наповнення;
- Дизайн курсу.

Причому термін «дизайн» тут використовується в двох значеннях. З одного боку, це методичний дизайн – структурування текстів, логічне вишикування їхніх частин, проектування структури поняттєво-категоріального апарату й інструментальної частини курсу – контрольних заходів, обговорень, критеріїв для оцінювання і т.д. Дуже важлива частина методичного дизайну – формування гіпертекстової структури курсу, тобто системи посилань і переходів між поняттями, змістовими й інструментальними блоками.

З іншого боку, дизайн – це «вбудовування» текстів, написаних викладачем і оброблених

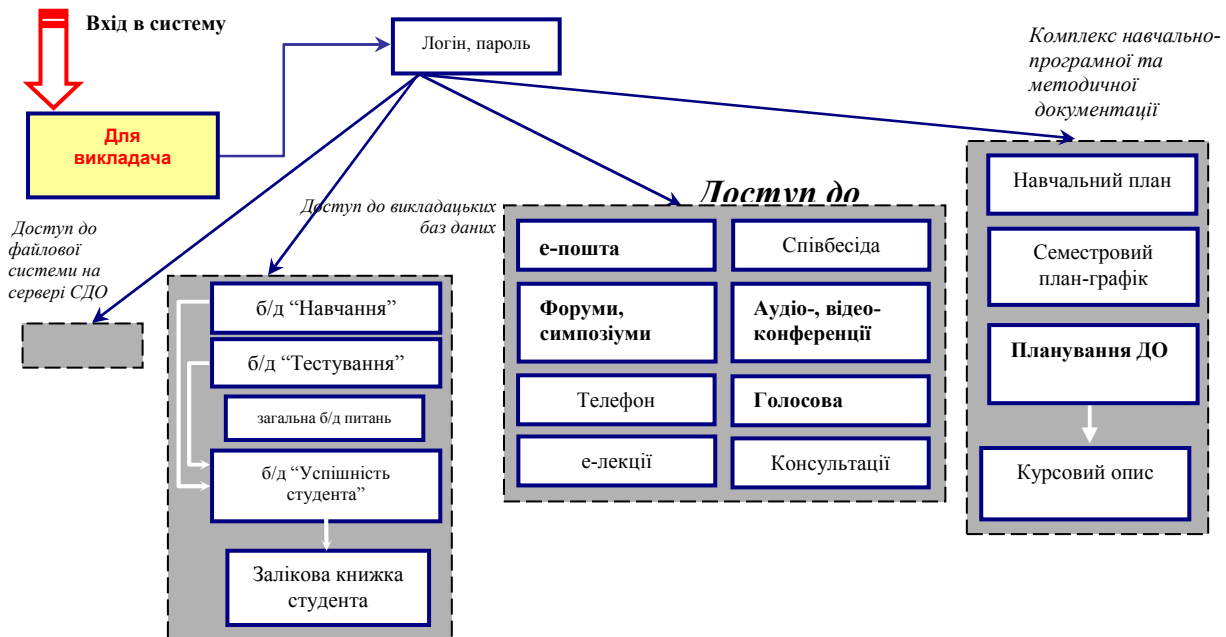


Рис. 3. Вхід в систему для викладача

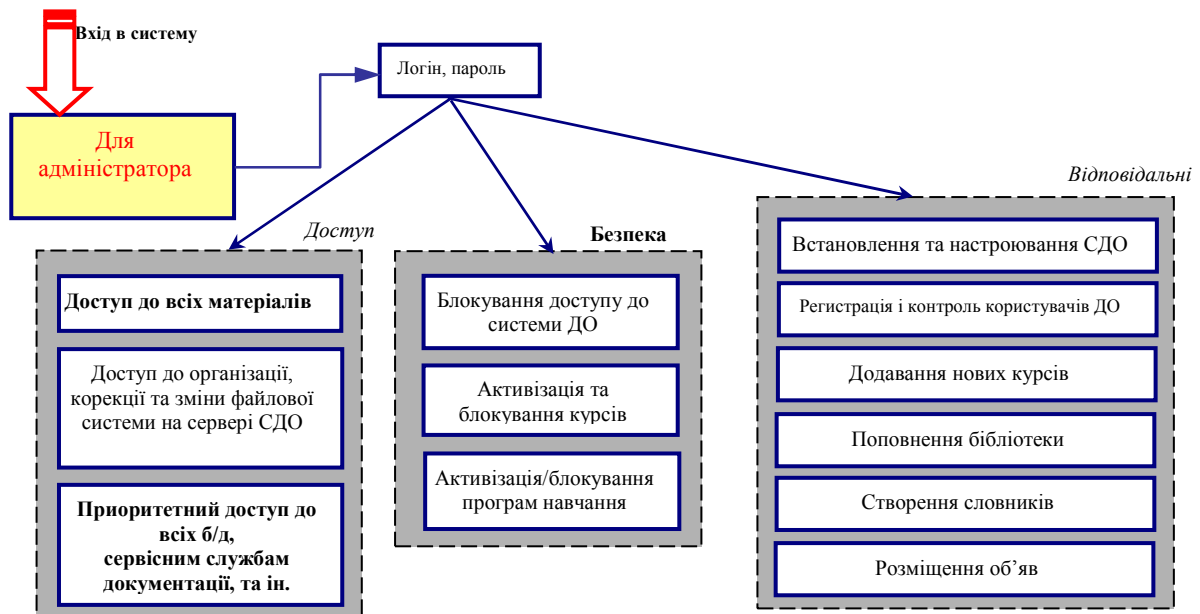


Рис. 4. Вхід в систему для адміністратора

методистом у стандартну веб-структуру, формування системи переходів і посилань. Ну і, зрозуміло, «наведення краси» – розробка і реалізація колірного вирішення, підбір ілюстрацій, розробка специфічних способів візуалізації, банерних модулів і т.д.

У даному розділі ми не будемо торкатись технічної сторони справи, а більше звернемо увагу на розробку самого курсу і його методичну обробку (тобто на дизайн у першому розумінні).

Отже, дистанційний курс являє собою особливим чином сконструйований *Web*-сайт (рис. 5), що складається з ряду сторінок-розділів.

Початкова сторінка (*home page*)

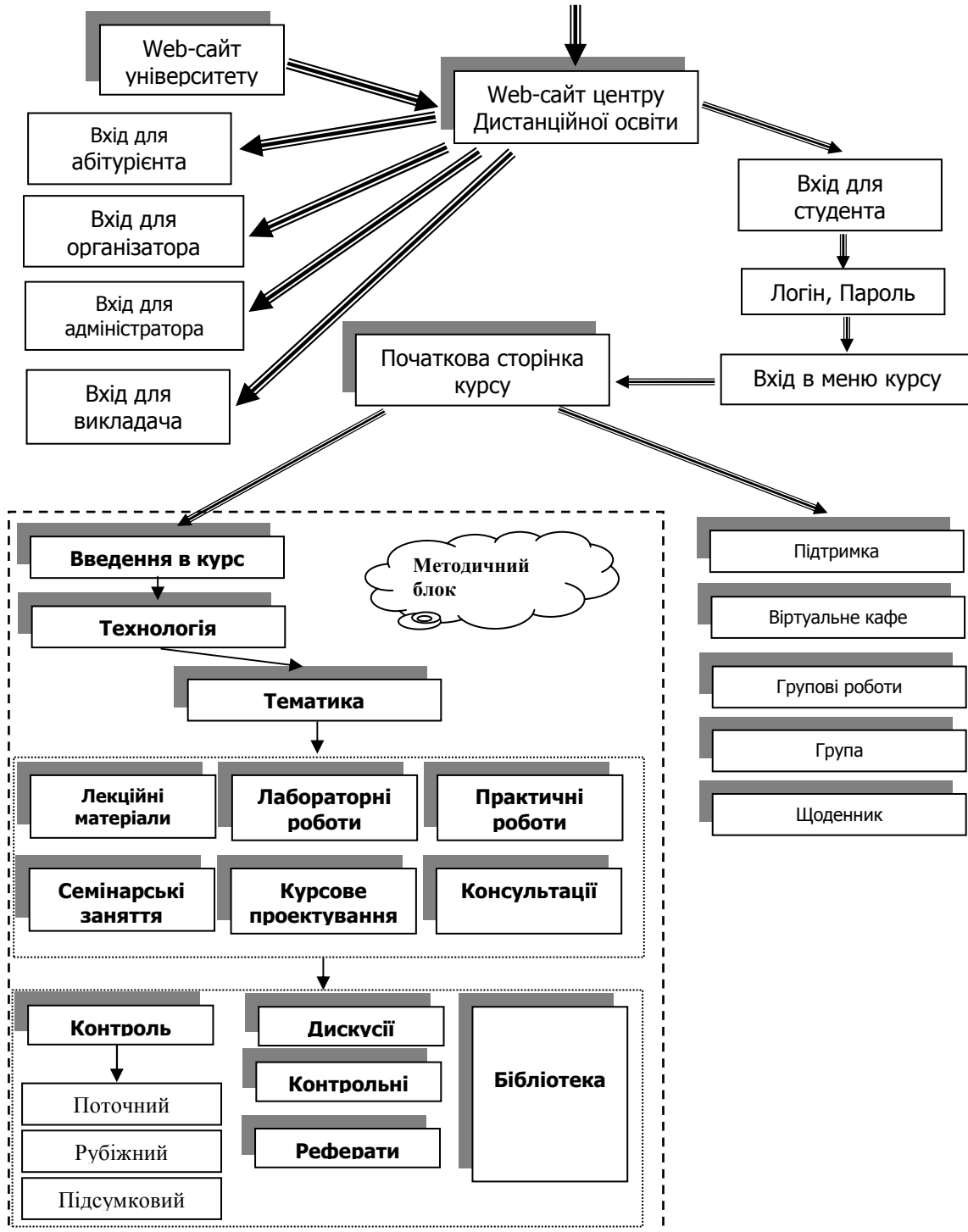


Рис. 5. Структура дистанційного курсу

Тут представлена загальна інформація про процес навчання. З цієї ж сторінки здійснюються переходи в інші частини курсу і, зокрема, на сторінку підтримки. Це сторінка, що містить консультації з типових проблем технічного і технологічного характеру (не пов'язаних зі змістом курсу). Вона найчастіше розташована поза курсовим сайтом, хоча, по суті, це не так вже й важливо.

Методичний блок

Фактично це найбільша частина курсу – саме те методичне наповнення, якому ми і приділимо далі особливу увагу. Цей блок містить у собі три базисних компоненти-сторінки: «Введення в курс», «Технологія» і «Тематика», а також похідні сторінки, вихід на які здійснюється з відповідних тем: «Контрольний блок», «Семінари» (чи – дискусійна сторінка), «Роботи» і «Бібліотека».

Організаційний блок

Це додаткові сторінки, що можуть вводитися зі збільшенням кількості студентів і технологічно-методичного удосконалення програми дистанційного навчання.

Група

Тут подається список студентів, що вивчають у поточному семестрі даний курс. Найчастіше тут дається об'єктивна інформація й елементи самопрезентації. Це не факультетська база даних (тобто не особиста справа), а, скоріше, груповий журнал.

Віртуальне кафе (чат або off-лайнова телеконференція)

Місце, де студенти можуть «побалакати» на теми, не пов'язані зі змістом курсу. Частіше, звичайно, таку функцію виконує один загальний чат, розташований поза окремим курсом – або на сайті університету, або на сайті відділення дистанційних технологій.

Групові проекти

Різновид дискусійної сторінки, призначений для спілкування студентів, що виконують груповий проект (якщо такий передбачений програмою курсу).

Щоденник

Такий інструмент у своїх дистанційних курсах використовують багато західних навчальних закладів. Це персональні міні-сторінки студентів, на яких вони описують власні цілі (що стосуються даного курсу), свій шлях в освоєнні курсу, якісь знахідки чи, навпаки, труднощі (об'єктивного і суб'єктивного характеру), з якими вони зіткнулися при вивченні курсу.

Висновки

На сьогоднішній день передавати знання, нехай дуже корисні – недостатньо. Є сенс передавати досвід, а саме, вміння використовувати знання на практиці, у реальному житті. Треба стимулювати людину вчитися протягом усього життя, і активно використовувати переваги ДО – віддаленість, свобода, якість отриманих знань.

Основна проблема полягає в тому, щоб якість масового навчання наблизити до індивідуального. Нові інформаційні технології дозволяють сподіватися на успіх. Однак ейфорія і бажання одержати негайний результат, спираючись тільки на інформаційні технології, швидше за все, призведуть до зворотного.

Отримавши визнання в Європі програма *ECTS (European Credit Transfer System)* може набирати оберти в тісній співпраці з дистанційною формою освіти.

Список літератури

1. Мокін Б.І., Мокін В.Б., Мокін О.Б. Дистанційні технології навчання – технології майбутнього освітньо-кваліфікаційних програм. Вінниця: ІОН. 2002. – с. 104-105.
2. Гороховський О.І., Трояновська Т. І. Структура електронного підручника. Вінниця – Каменец-Подольський: Наука и предпринимательство. – С. 309-312.

Гороховський Олександр Іванович, к.т.н., доцент, декан ФПК СКТ, кафедра ОТ, Вінницький національний технічний університет, Хмельницьке шосе, 95, Вінниця, 21021, Україна, тел.: (0432) 44-04-05, E-Mail: goroh@lili.vstu.vinnica.ua

Трояновська Тетяна Іванівна, студентка гр. 1-КС-00, Вінницький національний технічний університет, Хмельницьке шосе, 95, Вінниця, 21021, Україна.