

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ УКРАИНЫ

ХАРЬКОВСКИЙ НАЦИОНАЛЬНЫЙ УНИВЕРСИТЕТ
РАДИОЭЛЕКТРОНИКИ

ISSN 1563-0064

РАДИОЭЛЕКТРОНИКА И ИНФОРМАТИКА

Научно-технический журнал

Основан в 1997 г.

№ 4(79), октябрь – декабрь 2017

Выходит 4 раза в год

© Харьковский национальный
университет радиоэлектроники, 2017

Свидетельство о государственной регистрации КВ № 12097-968 ПР 14.12.2006

СОДЕРЖАНИЕ

ТЕЛЕКОММУНИКАЦИИ

СУЛИМА С.В. МЕТОД ВІДНОВЛЕННЯ МЕРЕЖІ У ВІРТУАЛІЗОВАНОМУ СЕРЕДОВИЩІ	4
ТАРАСЕНКО Д.А. МЕТОДОЛОГИЧЕСКИЕ АСПЕКТЫ ОБРАБОТКИ ДИНАМИЧЕСКОГО ВИДЕОРЕСУРСА ДЛЯ ПОВЫШЕНИЯ КАЧЕСТВА ДИСТАНЦИОННОГО ВИДЕОСЕРВИСА	9
ЛЕМЕСЬКО О.В., ЄРЕМЕНКО О.С. РОЗРОБКА ТА ДОСЛІДЖЕННЯ ЛІНІЙНОЇ ОПТИМІЗАЦІЙНОЇ МОДЕЛІ ШВИДКОЇ ПЕРЕМАРШРУТИЗАЦІЇ З БАЛАНСУВАННЯМ НАВАНТАЖЕННЯ В ТЕЛЕКОМУНІКАЦІЙНИХ МЕРЕЖАХ	18

СИСТЕМЫ И ПРОЦЕССЫ УПРАВЛЕНИЯ

ЧУБ І.А., НОВОЖИЛОВА М.В. МЕТОД ЗНИЖЕННЯ ЕКОЛОГІЧНОЇ НЕБЕЗПЕКИ ПРОМИСЛОВОГО ОБ'ЄКТА	26
---	----

КОМПЬЮТЕРНАЯ ИНЖЕНЕРИЯ

ЕМЕЛЬЯНОВ И.В., ЛЮБАРСКИЙ М.М., ХАХАНОВ В.И. КВАНТОВЫЕ МОДЕЛИ И ОБЛАЧНЫЕ СЕРВИСЫ ДЛЯ АНАЛИЗА И ДИАГНОСТИРОВАНИЯ ЛОГИЧЕСКИХ СХЕМ	30
ОБРИЗАН В.И., СОКЛАКОВА Т.Г. МУЛЬТИВЕРСНЫЙ ПАРАЛЛЕЛЬНЫЙ СИНТЕЗ ЦИФРОВЫХ СТРУКТУР НА ОСНОВЕ SYSTEMC СПЕЦИФИКАЦИИ	48

КОМПЬЮТЕРНЫЕ НАУКИ

ПРОХОРОВ А. В., ПРОХОРОВ В. П. АЛГОРИТМ УНИФИКАЦИИ В МЕТОДЕ РЕЗОЛЮЦИЙ ДЛЯ ЛОГИКИ ПРЕДИКАТОВ	53
ДОВБИШ А.С., ЗИМОВЕЦЬ В.І., КОЗЛОВ З.С. ІНФОРМАЦІЙНО-ЕКСТРЕМАЛЬНИЙ АЛГОРИТМ НАВЧАННЯ СИСТЕМИ ФУНКЦІОНАЛЬНОГО КОНТРОЛЮ ЕЛЕКТРОПРИВОДУ ШАХТНОЇ ПІДЙОМНОЇ МАШИНИ	58
ЗАЯЦЬ В.М., РИБИЦЬКА О.М., ЗАЯЦЬ М.М. ПРИХОВАНІ МОЖЛИВОСТІ МАТЕМАТИКИ ПРИ ОПРАЦЮВАННІ НЕЧІТКОЇ ІНФОРМАЦІЇ	64

ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ

ГВОЗДИНСКИЙ А.Н., СЕРИК Е.Э. ИССЛЕДОВАНИЕ МЕТОДОВ МНОГОКРИТЕРИАЛЬНОЙ ОЦЕНКИ УПРАВЛЕНИЯ ДЕЯТЕЛЬНОСТЬЮ СОВРЕМЕННОГО ПРЕДПРИЯТИЯ	72
ХАХАНОВ В.И., МИЩЕНКО А.С., СОКЛАКОВА Т.И., ЧУМАЧЕНКО С.В., ЛИТВИНОВА Е.И. ЛОГИЧЕСКИЕ МОДЕЛИ КИБЕРСОЦИАЛЬНОГО КОМПЬЮТИНГА	75
РЕФЕРАТИ	87
ПРАВИЛА ОФОРМЛЕНИЯ РУКОПИСЕЙ ДЛЯ АВТОРОВ НАУЧНО-ТЕХНИЧЕСКОГО ЖУРНАЛА	90

ТЕЛЕКОМУНІКАЦІИ

УДК 621.391+004.7

МЕТОД ВІДНОВЛЕННЯ МЕРЕЖІ У ВІРТУАЛІЗОВАНОМУ СЕРЕДОВИЩІ

СУЛИМА С.В.

Розглядається віртуалізація телекомунікаційної мережі. Описується метод динамічного відновлення вузлів з відмовою, який використовує евристичну модель для визначення оптимального місця міграції віртуалізованих функціональних вузлів з фізичного вузла з відмовою. Розглядається проблема вибору місць розташування децентралізованих вузлів управління.

Ключові слова: NFV, виділення ресурсів, відновлення мережі, мобільна мережа.

Key words: NFV, resource allocation, network recovery, mobile network.

1. Вступ

Хмарні обчислення – це нова парадигма, яка пропонує обчислювальні ресурси віртуалізованим способом з безпрецедентним рівнем гнучкості, надійності і масштабованості. У парадигмі хмарних обчислень передбачається, що обчислювальні ресурси пропонуються споживачам як сервіс. Це приводить до встановлення нової моделі обчислень, в якій такі ресурси управляються і виділяються інакше, ніж в традиційних сценаріях. При підключенні до хмарних інфраструктур можна отримати віддалений доступ до колекції віртуальних ресурсів, які включають в себе обробку, пам'ять, зберігання і передачу даних. Обчислювальні ресурси надаються динамічно, і очікується, що їх використання повинне дотримуватися угод про рівень обслуговування (Service Level Agreement – SLA) між провайдером інфраструктури і споживачами [1].

Оператори зв'язку все більше цікавляться віртуалізацією мережі. Деякими потенційними випадками використання в секторі телекомунікацій є простота розгортання вузлів, таких як MME, S/P-GW, масштабування за їх вимогою на основі миттєвого навантаження, а не на основі надмірного виділення ресурсів для пікового навантаження, динамічна реконфігурація топології для уникнення відмов і відновлення і т.д. Це прискорює розгортання мережі, зменшує зайняті ресурси, коли у цьому немає необхідності, і, таким чином, покращує ефективність використання ресурсів для збільшення доходу. Таке роз'єднання між програмним забезпеченням і апаратними засобами дозволяє також мігрувати віртуальні блоки. Дані властивості можна використовувати для перенесення критично важливих мережевих об'єктів при віртуалізації в безпечне місце під час стихійних лих [2].

У мережі з віртуалізацією мережевих функцій (Network Functions Virtualization – NFV) функції віртуальної мережі (Virtual Network Function – VNF) є програмним забезпеченням обробки трафі-

ка мережі, що працює на віртуальних машинах для обробки специфічних функцій мережі поверх апаратних засобів мережевої інфраструктури. Окремі функції віртуальної мережі можуть бути поєднані одна з одною як будівельні блоки, щоб запропонувати повномасштабний сервіс мережі зв'язку. VNF можна легко розгорнути або перерозподілити в будь-якому місці мережі.

Існуючі роботи, пов'язані з розгортанням VNF і міграцією, як правило, зосереджені на пропозиціях нових стратегій розгортання [3] і механізмів міграції [4]. Але вартість міграції не розглядається в цих дослідженнях. Насправді, вартість міграції є ключовим фактором в її процесі. Вона впливає на рішення про те, як вибрати кандидат VNF для перенесення і з'ясувати цільову позицію, яка підходить для міграції [5].

Стійкість обслуговування є важливою вимогою в будь-якій системі зв'язку, особливо в мобільних мережах. Крім того, доступність і надійність послуг, як зазначено в Рекомендації МСЕ-T E.800, визначають ключовий параметр характеристик якості обслуговування (QoS). В контексті хмарних мереж зв'язку забезпечувати стійкість сервісів стає проблемою. Дійсно, висока доступність є важливою вимогою, але не обов'язково є невід'ємною рисою хмарних обчислень. В хмарному середовищі на стійкість обслуговування може в значній мірі впливати відмова будь-якої VNF, що працює на віртуальній машині. Відмова VNF може статися через кілька факторів: відмова апаратних засобів (наприклад, через неправильне масштабування обладнання), вразливість програмного забезпечення та помилка в управлінні VNF або у її відповідній віртуальній машині, відмова на рівні гіпервізора через неправильну конфігурацію, негативний вплив на продуктивність за рахунок інших VNF, розміщених на одному фізичному вузлі, і зловмисні атаки проти VNF або менеджера віртуальної машини (тобто, гіпервізора) [6]. В операторській хмарі відмова VNF може вплинути на площину управління (наприклад, MME), а також на площину даних користувача (наприклад, S-GW або P-GW). Їх відмова істотно впливає на надання послуг, таким чином, важлива стійкість сервісів EPS шляхом визначення оперативних, масштабованих і надійних механізмів для відновлення після збоїв VNF [7].

Одна з основних проблем в контексті віртуалізації мережі полягає в тому, як ефективно використовувати фізичні ресурси (тобто центральний процесор, пам'ять вузлів і смугу пропускання каналів). В ряді досліджень [8,9] запропоновано різні методи для вирішення завдання відображення віртуальної мережі в надії встановлення ефективного використання фізичних ресурсів. У доповненні до методів, необхідних для ефективного відображення віртуальних мереж на фізичні мережі, потрібні методи, які керують ресурсами, вже виділеними активним

віртуальним мережам. На жаль, в літературі не описано таких методів. Отже, існує необхідність розробити методіку, яка може перемістити вже розміщені віртуальні вузли в разі відмови вузла при зведенні до мінімуму періоду переривання обслуговування.

У той час як адаптація шляхів для віртуальних мереж була розглянута у ряді робіт [8,10], проблема відмови вузла у віртуальних мережах була розглянуто лише у [11], проте не враховувалася вартість ресурсів на вузлі та кінцева якість обслуговування. Крім того, невирішеною залишалася задача вибору місць розташування вузлів управління.

У даній роботі представляється розподілений алгоритм відновлення вузла, який ефективно перерозподіляє віртуальні вузли, що постраждали від збою на фізичному вузлі. Основна мета пропонованої роботи полягає в розробці механізму самовідновлення віртуальної мережі, який може мінімізувати вартість відновлення вузла після відмови, а також підтримувати високий рівень фізичної працездатності мережі, що, в свою чергу, збільшує прибуток провайдера.

2. Модель мережі та постановка задачі

Фізична мережа задана у вигляді графа $SN=(N,L)$, де N є множиною фізичних вузлів і L – множиною каналів. Кожен канал $(n_1,n_2) \in L$, $n_1,n_2 \in N$ має максимальну пропускну здатність $cres(n_1,n_2)$ і мережеву затримку $L(n_1,n_2)$, а кожен вузол $n \in N$ пов'язаний з певними ресурсами $cres_n^i$, $i \in R$, де R – множина типів ресурсів. Мережа зв'язку представлена множиною ланцюгів сервісів (або запитів віртуальної мережі) T , які вбудовуються в фізичну мережу. Запит віртуальної мережі t , $t \in T$, можна представити як зважений граф $G_t=(V_t,E_t)$, де V_t є множиною віртуальних вузлів, що містить h_t елементів і позначається як $V_t=(v_{t,1},v_{t,2},\dots,v_{t,h_t})$, де $v_{t,j}$ означає j -у мережеву функцію у ланцюзі функцій t . E_t є множиною віртуальних каналів $e_t(v_{t,j},v_{t,g}) \in E_t$. Вимоги смуги пропускання каналу між двома функціями, j_1 і j_2 , що відносяться до ланцюга сервісів $t \in T$, позначаються як $d_t^{(j_1,j_2)}$, $d_t^{j,i}$ – кількість ресурсу типу i , що виділяється для мережевої функції j ланцюга t . Булеві змінні $x_n^{t,j}$ вказують, що мережева функція j , пов'язана з ланцюгом $t \in T$, розташовується на фізичному вузлі n , змінні $f_{(n_1,n_2)}^{t,(j_1,j_2)}$ визначають, що фізичний канал $(n_1,n_2) \in L$ використовується у шляху між j_1 та j_2 для запиту t . L_t – максимальна затримка для запиту $t \in T$; $costN(i,n)$ – вартість зайнятої одиниці ресурсу i на фізичному вузлі n , $i \in R$; $costL(n_1,n_2)$ – вартість зайнятої одиниці пропускну здатності на фізичному каналі $(n_1,n_2) \in L$; $suit_n^{t,j}$ означає, що функція j з запиту t може бути розміщена на вузлі n .

MN являє собою множину вузлів управління, де $MN \subseteq N$, які відповідають за функціонування пропонованого механізму відновлення після відмови. Кожен керуючий вузол пов'язаний з одним або де-

кількома вузлами фізичної мережі і виконує кроки, необхідні для відновлення після відмови мережі. Процес призначення вузлів управління і критерії, які враховуються при виборі вузлів управління, будуть досліджені далі.

У запропонованому підході припускаємо, що відображення запитів віртуальної мережі вже виконано (наприклад, з використанням підходу, визначеного в [9]).

Процес відображення віртуальної мережі відбувається в два етапи: відображення вузлів ($M_N: V_t \rightarrow N$) і відображення каналів ($M_L: E_t \rightarrow L$).

У пропонованому підході підкреслюється, що будь-яке переміщення віртуального вузла повинно виконуватись локально і бути координованим тільки вузлами управління.

3. Оптимальне розміщення вузлів управління у мережах, заснованих на NFV

Припускаємо, що вузли управління (далі – менеджери) можуть розміщуватись в точках, де оператор вже має сайт, тобто вузли управління розміщуються у вузлах N .

При заданій кількості менеджерів K існує скінчен-

на множина з $\binom{|N|}{K}$ можливих розташувань. Відпо-

відно, розміщення менеджерів є задачею багатокритеріальної комбінаторної оптимізації. Метою задачі є знаходження таких розташувань менеджерів з множини можливих розташувань розміру K –

$P_K = \left\{ P \in 2^{|N|} \mid |P| = K \right\}$, що є оптимальними відповідно до деякої цільової функції.

Метою оптимізації є визначення місця розташування кожного менеджера при заданій їх кількості K , так що мінімізується функція загальних витрат $Cost_K(\{p_k: k \in N\})$, де p_k – булева змінна, яка рівна одиниці, якщо менеджер розміщується в точці k . Задача оптимізації буде мати вигляд:

$$\begin{aligned} & \min_{\{p_k: k \in N\}} Cost_K \\ & \text{при обмеженні } \sum_k p_k = K \end{aligned} \quad (1)$$

Основною метою хорошого розміщення менеджерів є мінімізація затримок між вузлами і менеджерами в мережі. Проте розглядати тільки затримки не достатньо. Розміщення менеджерів повинно також враховувати певні обмеження стійкості [12]. На рис. 1 показані різні питання, які необхідно враховувати при оцінці стійкості розміщення. Показано також нормалізовані затримки між вузлами та інтенсивність навантаження на вузлах.

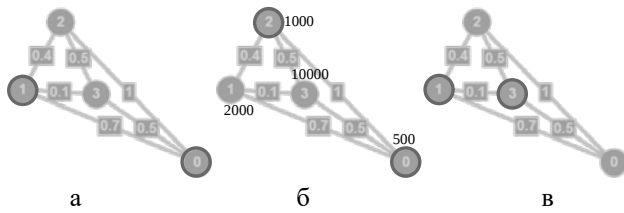


Рис. 1. Розміщення по різних критеріях: а -

мінімальної затримки до менеджера; б - мінімального дисбалансу навантаження на менеджерів; в - мінімальної затримки між менеджерами

Аналогічно до [12] припустимо, що вузли призначаються до їх найближчого менеджера, використовуючи як метрику затримки, тобто найкоротший шлях $dl_{g,k}$ між вузлом g і менеджером k . Кількість вузлів на менеджера може бути незбалансованою – чим більше вузлів менеджер повинен контролювати, тим вище навантаження на цього менеджера. Якщо кількість запитів вузла до менеджера в мережі збільшується, аналогічно поводить себе і ймовірність додаткових затримок через черги в системі управління. Для того щоб бути стійким від перевантаження менеджера, призначення вузлів різним менеджерам повинно бути добре збалансованим.

Цілком очевидно, що одного менеджера не досить, щоб досягти якої-небудь стійкості в мережі. Проте, коли кілька менеджерів розміщуються в мережі, логіка управління мережі розподіляється по декількох менеджерах і вони повинні синхронізуватися, щоб підтримувати несуперечний глобальний стан. Залежно від частоти синхронізації між менеджерами затримка між ними грає важливу роль.

На основі матриці dl , що містить відстані найкоротших шляхів між усіма вузлами, максимальний час затримки передачі між вузлом і менеджером для певного розміщення менеджерів може бути визначений як:

$$U_K^{\text{latency}}(p) = \max(ddc_k).$$

Тут ddc_k – максимальна затримка передачі від вузла мережі до менеджера в точці k ;

ddc_k розраховується так:

$$ddc_k = \max_{g \in V} \text{latency}_g \cdot \pi_{g,k},$$

де latency_g – затримка між менеджером та вузлом g , $\text{latency}_g = \min_{\{k: k \in V \wedge v_k = 1\}} dl_{g,k}$;

$\pi_{g,k}$ – булева змінна, яка рівна одиниці, якщо вузол g обслуговується менеджером розміщеним в точці k .

Розглядаємо не середнє, а максимальне значення затримки, оскільки середнє приховує значення найгіршого випадку, що є важливим, коли стійкість повинна бути поліпшена.

Залежно від ситуації бажано мати приблизно рівне навантаження на всіх менеджерів, так щоб жоден менеджер не перевантажувався. Далі розглядаємо збалансований розподіл вузлів між менеджерами. Як формальну метрику вводимо баланс розміщен-

ня або, вірніше, дисбаланс, $U_K^{\text{imbalance}}$, тобто відхилення від повністю збалансованого розподілу як різниці між навантаженням на найбільш і найменш завантаженому менеджері.

$U_K^{\text{imbalance}}$ визначається в такий спосіб:

$$U_K^{\text{imbalance}}(p) = \max(ldc_k) - \min(ldc_k), \text{ де } ldc_k > 0,$$

ldc_k – навантаження на менеджера в точці k ;

ldc_k розраховується так:

$$ldc_k = \sum_{g \in V} \text{load}_g \cdot \pi_{g,k},$$

де load_g – коефіцієнт навантаження на вузол g .

Як останній аспект стійкого розміщення менеджерів, розглянемо, як затримка між менеджерами може враховуватися при виборі розміщення менеджерів. Формально затримка між менеджерами $U_K^{\text{interlatency}}$ визначається як найбільша затримка між будь-якими двома менеджерами при заданому розміщенні:

$$U_K^{\text{interlatency}}(p) = \max_{\{k, g: k, g \in V \wedge p_k = 1, p_g = 1\}} dl_{g,k}.$$

Загалом, розміщення з урахуванням затримки між менеджерами мають тенденцію розміщувати всіх менеджерів набагато ближче один до одного. Це збільшує максимальну затримку від вузлів до менеджерів.

Таким чином, цільова функція оптимізації:

$$\text{Cost}_k = w_u^{\text{latency}} \times U_K^{\text{latency}}(p) + w_u^{\text{imbalance}} \times U_K^{\text{imbalance}}(p) + w_u^{\text{interlatency}} \times U_K^{\text{interlatency}}(p),$$

де w_u – множина вагових коефіцієнтів.

На рис. 2 показані можливі рішення по розміщенню двох менеджерів у мережі з 10 вузлами. Оптимальне значення показує, що усі оптимізаційні цілі не можуть бути досягнуті одночасно.

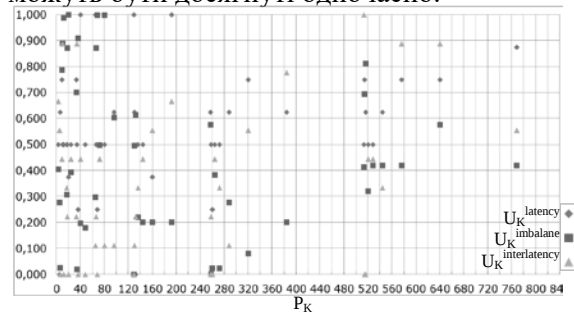


Рис. 2. Простір рішень оптимізаційної задачі розміщення менеджерів

4. Оптимальне розміщення вузлів управління у мережах, заснованих на NFV

Процес переміщення вузлів віртуальної мережі, розміщених на вузлі, який відмовив, $v_{t,j}^{\text{fail}}$, запускається, коли система відправляє запит на відновлення відповідному вузлу-менеджеру. Процес відновлення для кожного порушеного вузла віртуальної мережі протікає в такий спосіб: менеджер направляє запит на відновлення до всіх вузлів фізичної мережі, на яких розміщуються віртуальні вузли, суміжні з ураженими віртуальними вузлами. Кожен з цих вузлів будує дерево найкоротших шляхів (Shortest Path Tree – SPT) до всіх вузлів фі-

зичної мережі на відстані не більше 1 (поріг для максимального числа каналів встановлюється провайдером послуг) кроків від вузла, де коренем SPT виступає сам цей вузол. Менеджер використовує ці шляхи, щоб вибрати вузол з оптимальною відстанню до всіх вузлів фізичної мережі, де розташовані вузли віртуальної мережі, прилеглі до несправного вузла. Цей вузол в кінцевому рахунку стає оптимальним кандидатом для розміщення ураженого віртуального вузла. Довжина результатуючих шляхів з SPT буде не більше 1, максимальної кількості кроків, дозволеної для шляху, який відображає канал віртуальної мережі. Крім того, ємність кінцевих вузлів шляхів з SPT повинна бути не менша ємності віртуального вузла, розміщеного на несправному вузлі. Обираємо вузол з мінімальною вартістю шляху до всіх кореневих вузлів у деревах SPT та мінімальною вартістю обчислень. Алгоритм 1 (рис. 3) містить опис псевдокоду алгоритму відновлення вузла після відмови і виконується для всіх $\{v_{t,j} : x_n^{t,j}=1 \text{ \& } n=\text{failed}\}$.

```

 $x_n^{t,j} \leftarrow 0$ 
 $S_1 \leftarrow \{ m : \exists (e_i(j, m)) \}$ 
for all  $\{ m \in S_1 \}$  do
     $f^{(j,m)} \leftarrow 0$ 
     $w_m \leftarrow M_N(v_{t,m})$ 
end for
 $S_2 \leftarrow \bigcup_{m \in S_1} w_m$ 
Менеджер направляє запит SPT всім фізичним вузлам у  $S_2$ 
for all  $w \in S_2$  do
    Виконати алгоритм SPT
     $S_{3,w} \leftarrow \{ q : \text{length}(q, w) \leq l \}$ 
end for
 $S_4 \leftarrow \emptyset$ 
for all  $q \in \bigcup_{w \in S_2} S_{3,w}$  do
    for all  $\{ m \in S_1 \}$  do
        if  $\exists (e_i(j, m))$  then
             $f_{(q,w_m)}^{t,j,m} \leftarrow 1$ 
        end if
    end for
    if  $(\sum_{(b_1,b_2) \in E_t} \sum_{(a_1,a_2) \in L} f_{(a_1,a_2)}^{t,(b_1,b_2)} \cdot L(a_1, a_2) \leq L_t \text{ \& \& } d_t^{j,i} \leq \text{cres}_q^i \forall i \in R \text{ \& \& } d_t^{(j,m)} \leq \text{cres}(q, w_m) \forall m \in S_1)$  then
         $\text{CostNL}_q \leftarrow \text{weight}_1 \cdot \sum_{i \in R} d_i^{t,j} \cdot \text{costN}(i, q) + \text{weight}_2 \cdot \sum_{w \in S_2} \text{costL}(q, w_m) \cdot d_t^{(j,m)}$ 
    end if
     $S_4 \leftarrow S_4 \cup q$ 
end if
for all  $\{ m \in S_1 \}$  do
     $f^{(j,m)} \leftarrow 0$ 
end for
end for
if  $S_4 = \emptyset$  then
    Виконати алгоритм Реконфігурації
else
    Обрати  $\min \text{CostNL}_q, q \in S_4$ 
     $q^* = \text{argmin CostNL}$ 
end if
 $x_{q^*}^{t,j} \leftarrow 1$ 
for all  $\{ m \in S_1 \}$  do
    if  $\exists (e_i(j, m))$  then
         $f_{(q^*,w_m)}^{t,j,m} \leftarrow 1$ 
    end if
end for
end for

```

Рис. 3. Алгоритм відновлення вузла з відмовою

5. Висновки

Щоб впоратися зі значним зростанням трафіка мобільних даних, мобільний оператор освоює технології віртуалізації мережі та хмарних обчислень, щоб побудувати ефективні та гнучкі мобільні мережі та пропонувати їх як хмарний сервіс. У роботі

встановлено, що динамічне надання віртуалізованих ресурсів у системах мобільного зв'язку ставить нові задачі, не вирішені у попередніх дослідженнях систем надання ресурсів. У статті розглянута проблема відновлення віртуалізованої мережі, що постраждала від відмови на нижчезрозташованому фізичному вузлі. Пропонований підхід виконується розподіленим способом. У дослідженні вирішена проблема розташування локальних вузлів управління з урахуванням параметрів затримки та стійкості, запропонована цільова функція комбінаторної оптимізаційної задачі розташування вузлів управління. Запропоновано метод динамічного відновлення вузлів з відмовою, який використовує евристичну модель для визначення оптимального місця міграції віртуалізованих функціональних вузлів.

Метод може застосовуватись при управлінні функціонуванням віртуалізованої мережі зв'язку для мінімізації витрат оператора зв'язку та покращення якості обслуговування абонентів. В подальших дослідженнях запропонований метод може бути розширений для урахування випадку відмови вузла через його перевантаження та можливості реконфігурації всієї системи.

Література:

1. Castro P.H.P. A joint CPU-RAM energy efficient and SLA-compliant approach for cloud data centers / P.H.P. Castro, V.L. Barreto, S.L. Corrêa, L.Z. Granville, K.V. Cardoso // Computer Networks. 2016. Vol. 94. P. 1-13.
2. Khan A. Virtual Network Embedding for telco-grade network protection and service availability/ A. Khan, X. An, S. Iwashina // Computer Communications. 2016. Vol. 84. P. 25-38.
3. Xiong G. A virtual service placement approach based on improved quantum genetic algorithm / G. Xiong, Y.-X. Hu, L. Tian, J.-L. Lan, J.-F. Li, And Q. Zhou // Frontiers of Information Technology & Electronic Engineering. 2016. Vol. 17, No. 7. P. 661–671.
4. Gember-Jacobson A. OpenNF: Enabling Innovation in Network Function Control / A. Gember-Jacobson, R. Viswanathan, C. Prakash, R. Grandl, J. Khalid, S. Das, and A. Akella, // Proceedings of the 2014 ACM Conference on SIGCOMM, ser. SIGCOMM '14. New York, NY, USA, 2014. P. 163–174.
5. Xia J. Optimized Virtual Network Functions Migration for NFV/ J. Xia, Z. Cai, M. Xu // IEEE 22nd International Conference on Parallel and Distributed Systems. Wuhan, China, 2016. P. 340-346.
6. Scholler M. Resilient deployment of virtual network functions / M. Scholler, M. Stiernerling, A. Ripke, R. Bless // Proc. 5th Int. Congr. Ultra Mod. Telecommun. Control Syst. Workshops (ICUMT), 2013. P. 208-214.
7. Taleb T. On Service Resilience in Cloud-Native 5G Mobile Systems / T. Taleb, A. Ksentini, B. Sericola // IEEE Journal on Selected Areas in Communications. 2016. Vol. 34, No. 3. P. 483-496.
8. Fajjari I. VNR Algorithm: A Greedy Approach For Virtual Networks Reconfigurations / I. Fajjari, N. Aitsaadi,

G. Pujolle and H. Zimmermann // IEEE Global Communications Conference, Exhibition and Industry Forum, Houston, United States, 2011.

9. *Skulysh M.* Model for Efficient Allocation of Network Functions in Hybrid Environment / M. Skulysh, L. Globa, S. Sulima // Information and Telecommunication Sciences. 2016. № 1. P. 39–45.

10. *Rahman M.R.* Survivable Virtual Network Embedding / M. R. Rahman, I. Aib, and R. Boutaba // IFIP International Federation for Information Processing, 2010.

11. *Abid H.* A novel scheme for node failure recovery in virtualized networks / H. Abid; N. Samaan // 2013 IFIP/IEEE International Symposium on Integrated Network Management (IM 2013), Ghent, Belgium, 2013. P. 1154-1160.

12. *Hock D.* Pareto-optimal resilient controller placement in SDN-based core networks / D. Hock, M. Hartmann, S. Gebert, M. Jarschel, T. Zinner, P. Tran-Gia // 25th International Teletraffic Congress (ITC), Shanghai, 2013. P. 1-9.

Надійшла до редколегії 09.09.2017

Рецензент: д-р техн. наук, проф. Безрук В.М.

Суліма Світлана Валеріївна, аспірантка Національного технічного університету України "Київський політехнічний інститут". Наукові інтереси: мобільні мережі, NFV. Адреса: Україна, Київ, пров. Індустріальний, 2, тел. 0666245361.

Sulima Svitlana Valeriivna, Ph.D. student at National technical university of Ukraine "Kiev Polytechnic Institute". Mobile networks, NFV. Address: Ukraine, Kyiv, pr. Industrialnyy 2, 0666245361.

МЕТОДОЛОГИЧЕСКИЕ АСПЕКТЫ ОБРАБОТКИ ДИНАМИЧЕСКОГО ВИДЕОРЕСУРСА ДЛЯ ПОВЫШЕНИЯ КАЧЕСТВА ДИСТАНЦИОННОГО ВИДЕОСЕРВИСА

ТАРАСЕНКО Д.А.

Обосновывается, что для повышения эффективности управления ведомственными организациями и профильными министерствами, в том числе в кризисных условиях, возрастает роль и значимость дистанционных видеoinформационных сервисов с использованием аэромобильных платформ. Доказывается наличие дисбаланса между требованиями пользователей относительно качества предоставляемого динамического видеоресурса и недостаточными возможностями аэромобильных платформ относительно производительности аппаратуры обработки и передачи информации.

Ключевые слова: динамический видеоресурс, кодирование видеопотока, дистанционный видеосервис, кадр Р-типа

Keywords: dynamic video resource, video stream encoding, remote video service, P-frame.

1. Введение

Экономический рост, социальное благополучие населения и национальная безопасность государства определяется уровнем развития наукоемких технологий, включая информационные технологии информационного предоставления видеoinформационных сервисов [1-10].

В этом случае источниками информации (сенсорами) в зависимости от своего базирования могут быть источники наземного и аэрокосмического базирования. При этом за последнее время ключевым технологическим фундаментом информационного обеспечения и поддержки принятия решений становится видеoinформация или видеoinформационные ресурсы.

В рамках категории общегосударственного значения важность видеoinформационных сервисов подчеркивается необходимостью обеспечить безопасность и повысить качество функционирования и управления [5 - 10]:

- 1) стратегическими объектами, в первую очередь теми, которые относятся к объектам критической инфраструктуры: атомные и гидроэлектростанции; нефтехимические производства, металлургические предприятия, нефтегазовая транспортная система;
- 2) мероприятиями международного значения;
- 3) локализацией и ликвидацией кризисных ситуаций естественного и техногенного характеров: наводнения; пожары лесных массивов; взрывоопасные ситуации на арсеналах вооружения и военной техники.

Развитию видеoinформационного взаимодействия способствует повышение качества, рост производительности и доступность персональных информационных технологий, к числу которых относятся беспроводные мобильные терминалы. Это объясняет необходимость усовершенствования видеoinформационных сервисов как для мобильных пользователей, так и для стационарных абонентов с использованием существующих и перспективных беспроводных инфокоммуникационных технологий. Данный аспект составляет научно-прикладную задачу исследований.

2. Оценка дисбаланса в процессе организации дистанционных видеосервисов

В процессе предоставления видеосервисов в интересах профильных организаций, в том числе в процессе функционирования в условиях кризисных ситуаций, необходимо учитывать такие факторы:

- 1) требование относительно получения информации одновременно с больших по масштабу территорий;
- 2) наличие труднодоступных для наземного транспорта территорий, в том числе горных и лесных массивов, морских и речных акваторий;
- 3) наличие быстрого распространения кризисных ситуаций на значительные территории.

В связи с этим, организация предоставления видеосервисов в интересах профильных организаций должна использовать дистанционные сенсорные платформы как наземного, так и аэрокосмического базирования.

При этом актуальным является направление, основанное на использовании дистанционных бортовых платформ (комплексов) воздушного базирования (аэромобильных платформ) [11 – 19]. Аэромобильные платформы строятся с использованием воздушных средств: беспилотные летательные аппараты, пилотируемая авиация, включая самолеты и вертолеты.

Это обосновывается наличием целого ряда преимуществ мониторинга на основе аэромобильных платформ по сравнению с платформами космического сегмента, а именно: снижением финансовых затрат; возможностью оперативного управления летательным аппаратом либо дистанционно (беспилотники), либо пилотируемо (вертолеты); возможностью быстрого обновления информации о контролируемых объектах (не требуется ждать, пока будет сделан оборот спутника).

Видеoinформационные сервисы на основе дистанционных аэромобильных платформ обеспечи-

вают пользователей двумя основными типами видеoinформации:

- 1) статический видеоресурс, включающий аэрофотоснимки таких классов как: панорамные, наблюдательные, объективного контроля;
- 2) динамический видеоресурс, представляющий собой поток видеокадров.

Динамический видеоресурс является основным типом информации, которая предоставляется видеoinформационными сервисами для пользователей, включая видеоконференцсвязь, цифровое ТВ высокого качества, видеомониторинг местности, телемедицина, охранное видео, виртуальная реальность, техническое зрение.

Анализ информации указывает на развитие видеoinформационных сервисов, предоставляющих интеллектуализированные услуги в реальном времени. При этом повышается спрос на динамическое видео – информационный видеопоток.

Динамический видеоресурс (видеопоток) характеризуется двумя типами разрешающей способности:

- 1) структурное разрешение;
- 2) семантическое разрешение.

Структурная разрешающая способность видеопотока определяется такими параметрами как: количество кадров, генерируемых сенсорной системой в единицу времени (частота кадров) – разрешение по времени; количество строк и столбцов в кадре – внутрикадровая разрешающая способность по строкам и столбцам (размерность кадра); количество бит, используемых для оцифровки значения пикселя (отдельного элемента видеокадра) – пиксельное разрешение.

Семантическая разрешающая способность видеопотока определяется типом задач распознавания, решение которых необходимо обеспечить с использованием видеoinформационного сервиса. Семантическая разрешающая способность определяется параметром детальности видеокадров. Требуемая разрешающая способность видеокадров для видеопотока в интересах профильных министерств определяется из условия обеспечения необходимого уровня детализации, влияющего на процесс распознавания типового объекта контроля или объекта визуализации, например, для систем технического зрения. Оценка уровня детализации или характерной детальности на объект визуализации с учетом уровня распознавания типового объекта определяется на основе эмпирического подхода с использованием методики «миры», или по заданным табличным значениям.

Обобщающими характеристиками видеопотока являются информационная интенсивность в условиях заданного уровня целостности и семантическая разрешающая способность. Информационная интенсивность для заданной детализации с учетом трехцветной модели цифрового кадра определяется как суммарное количество бит, формируемое видеосенсором за единицу времени.

Системной проблемой организации дистанционных видеосервисов является то, что существуют значительные временные задержки, которые резко увеличиваются в случае передачи динамического видеоресурса (видеопотока) в реальном времени.

Такие временные задержки в свою очередь влияют на снижение качества других показателей, включая:

- вероятность потери пакетов на узле доступа из-за переполнения буферной памяти;
- загрузку узла доступа, показывающую задействованную вычислительную способность узла доступа при обработке принятых пакетов;
- пропускную способность, которая отражает объем данных, переданных сетью в единицу времени;
- задержку от источника до получателя, которая определяется как время между возникновением запроса к какому-либо сетевому сервису и получением ответа на него.

Значит, информационная интенсивность значимым образом влияет на качество предоставления видеoinформационных сервисов с использованием платформ на базе БЛК. Основная причина здесь состоит в наличии дисбаланса между: с одной стороны, требованиями пользователей относительно качества предоставляемого динамического видеоресурса, а это в первую очередь рост интенсивности видеопотока в условиях заданного уровня целостности информации, а с другой стороны – недостаточными возможностями аэромобильных платформ относительно производительности аппаратуры обработки и передачи информации.

Поэтому разработка методологических аспектов обработки динамического видеоресурса для повышения качества дистанционного видеосервиса является целью исследований данной статьи.

3. Разработка методологических аспектов обработки динамического видеоресурса

Эффективность функционирования информационных технологий ИТОПК в системах предоставления дистанционных видеoinформационных сервисов с использованием аэромобильных средств обеспечивает:

- 1) снижение вероятности потери пакетов и временной задержки на доставку видеопотока с за-

данным уровнем детализации ρ_0 объектов интереса;

2) повышение уровня информативности потока кадров и качества видеозображений;

3) повышение времени, предоставляемого для анализа и принятия решений ЛПР (оператором, диспетчером);

4) создание условий для повышения качества автоматизированного анализа и идентификации объектов для динамических видеоресурсов, т.е. создаются условия для роста уровня интеллектуализации видеоинформационных сервисов;

5) уменьшение требований относительно производительности бортовых инфокоммуникационных технологий, что создает условия для повышения уровня энергоэффективности использования бортовых беспилотных комплексов (рост сеанса связи, увеличение рабочего времени для мониторинга);

6) создание условия для одновременной передачи информации, полученной от других бортовых сенсоров (радиолокационных, инфракрасных);

7) снижение риска потери беспилотного бортового комплекса.

Современные стандартизированные информационные технологии обработки и передачи кадров строятся на основе использования MPEG-платформ и технологий семейства H26x, включая технологии H264, H265. Основное назначение таких технологий состоит в снижении информационной интенсивности $V(\rho_0)_t$ динамических видеоресурсов с заданной детализацией ρ_0 объектов интереса в условиях обеспечения компромиссных решений относительно целостности информации. Для этого стандартизированные ИТОПК используют концепцию классификации кадров в локальных группах в зависимости от обеспечения весовой составляющей в суммарный баланс между интенсивностью $V(\rho_0; t)_k$ кодированного видеопотока и целостностью информации.

В общем случае здесь дифференцируются три весовые составляющие.

Первая образуется базовыми кадрами - кадрами I-типа. Здесь весовая составляющая $V(I)_k$, приходящаяся на один кадр в группе, является максимальной. Для кадров данного типа обеспечивается наибольшее сохранение целостности информации. Относительно базового кадра напрямую или опосредовано формируются кодовые представления остальных кадров в группе – кадры Р-типа и В-типа.

Количество v_{grp} кадров в группе согласно стандартизированным спецификациям выбирается

равным 8 или 12. При этом количество $v(P)$ кадров Р-типа и количество $v(B)$ кадров В-типа принимается соответственно порядка 30 – 40 % и 60 – 70 % от общего количества кадров в группе.

Вторая весовая составляющая образуется кадрами Р-типа. Кадры такого класса определяются как предсказываемые или напрямую относительно базового кадра (кадра I-типа), или опосредовано через промежуточные кадры Р-типа. Собственно на кадры Р-типа возлагается ключевая роль относительно обеспечения баланса между интенсивностью кодированного видеоресурса и его целостностью. Это достигается в результате:

– снижения информационной интенсивности, приходящейся на кадры Р-типа ценой внесения искажений;

– регулирования их количества $v(P)$ в группе.

В этом случае весовая составляющая $V(P)_k$, приходящаяся на один кадр Р-типа в группе, имеет промежуточный уровень между весовой составляющей $V(I)_k$ базового кадра и $V(P)_k$ кадра В-типа.

Третья весовая составляющая формируется кадрами В-типа. Данные кадры определяются в результате двунаправленного предсказания относительно смежных кадров Р-типа и В-типа. Количество кадров В-типа позволяет управлять снижением интенсивности $V(\rho_0; t)_k$ суммарного кодированного видеопотока в зависимости от изменений пропускной способности бортовых телекоммуникационных технологий. Увеличение количества кадров В-типа объясняется снижением времени обработки всей группы кадров и уменьшением интенсивности видеопотока. В то же время кадры данного типа характеризуются наименьшим уровнем сохранения целостности исходного видеоресурса. Поэтому рост их количества в группе кадров приводит к потере целостности информации (снижается значение h пикового отношения сигнал/шум).

В общем случае информационная интенсивность $V(\rho_0; t)_k$ кодированного видеопотока за время t

через усредненную интенсивность $V_{grp}^{(v_{grp})}$ группы кадров определяется по следующей формуле:

$$V(\rho_0; t)_k = \frac{v_t}{v_{grp}} V_{grp}^{(v_{grp})} \text{ (бит)}. \quad (1)$$

Здесь величина $\frac{v_t}{v_{grp}}$ определяет количество групп в видеопотока за время t .

Соответственно информационная интенсивность $V_{\text{grp}}^{(\text{vgrp})}$ видеопотока, приходящаяся на группу кадров, определяется с помощью такого выражения:

$$V_{\text{grp}}^{(\text{vgrp})} = V(I)_k + v(P) V(P)_k + v(B) V(B)_k. \quad (2)$$

Понятно, что от степени снижения информационной интенсивности кадра и временных затрат на обработку будет зависеть уровень вносимых искажений в его синтаксическое и семантическое содержание. Значит, величина пикового отношения сигнал/шум будет также иметь дифференцированные значения в зависимости от типа кадра в группе.

Для обеспечения требуемого уровня целостности динамического видеоресурса для каждого типа кадра в зависимости от его роли в общем процессе сохранения информации существует определенный уровень ПОСШ. На выбор минимально допустимого уровня ПОСШ каждого типа кадра группы влияют такие факторы:

- степень воздействия кадра на процесс восстановления других кадров, т.е. уровень информационной взаимосвязи с другими кадрами;
- класс и глубина задач по идентификации и анализу объектов аэромобильного мониторинга, т.е. степень сохранения исходной детализации объектов на видеокадре;
- уровень автоматизации процесса дешифрирования объектов для динамического видеоресурса;
- требования по оперативности обработки и анализа информации для принятия решений;
- стратегия обеспечения баланса между интенсивностью видеопотока и его целостностью с учетом характеристик производительности инфокоммуникационных технологий, базирующихся на бортовом беспилотном комплексе дистанционного мониторинга.

Чем выше степень информационного влияния типа кадра в процессе сохранения целостности информации для реконструкции других кадров, тем выше минимально допустимый уровень ПОСШ. Отсюда наибольший уровень ПОСШ необходимо обеспечить для базового кадра. Соответственно для кадра Р-типа требуется обеспечить наибольшее снижение информационной интенсивности при заданном уровне ПОСШ.

Для потока видеокадров с высоким структурным и семантическим разрешением, чтобы обеспечить своевременную доставку с использованием перспективных беспроводных инфокоммуникационных технологий с пропускной способностью не ниже 100 Мбит/с, в условиях ПОСШ на уровне не

ниже 30 дБ, необходимо величину информационной интенсивности дополнительно снизить как минимум от 1,7 до 3,5 раза.

4. Обоснование базовых компонент внутрикадрового синтаксического кодирования видеопотока для информационных технологий обработки и передачи информации

Совершенствование информационных технологий обработки и передачи потока кадров (ИТОПК) предлагается осуществлять в направлении повышения эффективности кодирования последовательности кадров Р-типа.

Особенность кадров Р-типа заключается в том, что они содержат изменения относительно предыдущего I- или Р-кадра. В зависимости от кодировщика выделяются Р-кадры либо с фиксированной частотой (чтобы каждый I- или Р-кадр был окружён двумя В-кадрами, например, IBVRBVRBVRBVRB...), либо автоматически (по сложности видеоряда). В число всех макроблоков для ДКП в этих кадрах входит небольшое количество независимых (неподдающихся предсказанию) блоков. Основную же часть кадра составляют предсказанные макроблоки. При воспроизведении, чтобы восстановить кадр Р-типа, необходимо восстановить все предшествующие ему кадры данного типа и ближайший предшествующий базовый кадр (кадр I-типа). В последствии результат кодирования каждого кадра Р-типа подвергается обработке методами статистического компрессионного кодирования и вставляется в поток отдельно от кадров I-типа и других кадров Р-типа. Это связано с необходимостью быстрых переходов к произвольному кадру и перестройки последовательности кадров в видеопотоке.

Ключевым технологическим ядром здесь является программно-аппартный блок, реализующий эффективное внутрикадровое синтаксическое представление данных кадра Р-типа.

В настоящее время для информационных технологий ИТОПК используются несколько десятков кодеков [12 - 19]. Современные технологии внутрикадрового синтаксического кодирования с компрессией являются комплексными. Такие технологии основываются на совокупности технологических механизмов, обеспечивающих:

- 1) выявление различных закономерностей и сокращение соответствующих видов избыточности в условиях заданного уровня целостности информации;
- 2) реализацию протоколов передачи информации на различных уровнях инфокоммуникационных систем в соответствии с моделью OSI;

3) реконструкцию кадров Р-типа для видеопотока с учетом зависимости их контента от базового кадра.

На практике реализация технологических этапов, связанных с внутрикадровым кодированием, организуется на платформах семейства JPEG. Здесь характерными аспектами являются:

- 1) возможность проведения обработки кадров Р-типа в режимах без потери целостности информации и в режиме наличия искажений их синтаксического представления;
- 2) возможность снижения нескольких видов избыточности как в пространственно-временном, так и в пространственно-частотном промежуточном синтаксическом представлении.

Одним из подходов относительно реализации технологии синтаксического кодирования трансформант является их обработка в области компонентного описания. Для стандартизированных ИТОПК такая обработка строится на основе базиса выделения значимых компонент $z(\tau; \delta)_u$ и соответственно длин $l(\tau; \delta)_u$ последовательностей незначимых компонент трансформанты. Такой этап, связанный с выявлением длин последовательностей незначимых компонент, обеспечивает сокращение структурно-статистической избыточности.

После выявления значимых компонент и соответственно последовательностей незначимых компонент в рамках реализации синтаксического кодирования для стандартизированных ИТОПК формируются двухкомпонентные векторы, а именно $\{l(\tau; \delta)_u; z(\tau; \delta)_u\}$. Здесь $l(\tau; \delta)_u$ - длина последовательности незначимых компонент, предшествующих значимой компоненте $z(\tau; \delta)_u$ для τ -й трансформанты в случае ее квантования с параметром δ .

Формирование двухкомпонентных векторов $\{l(\tau; \delta)_u; z(\tau; \delta)_u\}$ - всего лишь промежуточный этап общего процесса синтаксического представления трансформанты, предусмотренного стандартизированными ИТОПК. Одним из ключевых технологических этапов является стратегия выделения количества разрядов под компоненты $l(\tau; \delta)_u$ и $z(\tau; \delta)_u$. Такой этап называется кодообразованием. Эффективность кодообразования зависит от адекватности выбранной модели информативности синтаксического описания величин $l(\tau; \delta)_u$ и $z(\tau; \delta)_u$ реальному содержанию обрабатываемой трансформанты. Если модель информативности выбрана не адекватно, то гипотеза о наличии предполагаемых закономерностей для двухкомпо-

нентных векторов не подтверждается. Это значит, что выявляемые закономерности в трансформанте либо слабо выражены, либо вообще отсутствуют. Соответственно созданный механизм кодирования не приведет к сокращению информационной интенсивности сегмента. Наоборот, адекватно выбранная модель информативности обеспечивает выявление соответствующих закономерностей. Это создает условия для сокращения избыточности в процессе кодирования.

Для определения базовой модели оценивания информативности рассмотрим синтаксическое описание трансформанты, задаваемое двумерным массивом $Y_2^{(1)}: Y^{(1)} \rightarrow Y_2^{(1)} = \{l(\tau; \delta)_u; z(\tau; \delta)_u\}$, $u = \overline{2, U-1}$.

Данный массив рассматривается как двумерный структурно-комбинаторный объект, строками которого являются перестановки с повторениями по $U-2$ элементов с двумя спецификациями, определяемыми как ограничения на динамический диапазон соответственно $d(l; \delta)_\tau$, $d(z; \delta)_\tau$.

С одной стороны, такой подход относительно формирования эффективного внутрикадрового синтаксического кодирования обладает потенциалом для дополнительного снижения интенсивности потока кадров. С другой стороны, он не в полной мере учитывает особенности обработки кадров Р-типа.

Поэтому для дополнительного снижения информационной интенсивности предлагается учитывать новые структурные закономерности, которые создаются в процессе обработки трансформированных сегментов кадров Р-типа. В рамках этого предлагается осуществлять перетрансформацию структурного пространства трансформанты $\{L; Z\}$. Суть перетрансформации векторов $\{l(\tau; \delta)_u; z(\tau; \delta)_u\}$ заключается в обеспечении тенденции пропорционального изменения значений его компонент $l(\tau; \delta)_u$ и $z(\tau; \delta)_u$. Для этого предлагается значимые компоненты линейаризованной трансформанты размещать в обратном порядке. В этом случае перетрансформация ДК описывается следующим выражением:

$$\tilde{r}_u : \{l(\tau; \delta)_u; z(\tau; \delta)_{U-u+1}\}, \quad (3)$$

где $\tilde{r}_u$ - u -я перетрансформированная дискретная позиция двумерного структурного пространства (ДСП) $\{L; Z\}$ трансформанты.

В результате этого строится перетрансформированный вектор $\tilde{R}(U-2)$ дискретных позиций ДСП пространства, который в координатной форме

примет следующий вид:
 $\tilde{P}(U-2)=\{\tilde{p}_2; \dots \tilde{p}_u; \dots \tilde{p}_{U-1}\}.$

С учетом этого предлагается формировать синтаксическое представление трансформанты Y в двумерном структурном пространстве $\{L; Z\}$ по объектному принципу. Здесь единое кодовое значение строится для координат $\{l(\tau; \delta)_u; z(\tau; \delta)_{U-u+1}\}$ дискретной позиции $\tilde{p}_u$ ДСП пространства, как для отдельных объектов. Такие кодовые значения предлагается называть идентификаторами $I(\tau; \delta)_u$, что задается выражением:
 $\tilde{p}_u: \{l(\tau; \delta)_u; z(\tau; \delta)_{U-u+1}\} \rightarrow I(\tau; \delta)_u.$

При этом требуется учитывать следующие аспекты:

1. Количество дискретных отсчетов ДСП пространства является неравномерным контекстно-зависимым, т.е. $U = \text{var}.$
2. Двумерное структурное пространство $\{L; Z\}$ является уплотненным по значимым компонентам спектра, что задается неравенством $U < n^2.$
3. Максимальные значения элементов векторов L и Z ограничены соответственно величинами $d(l; \delta)_\tau$ и $d(z; \delta)_\tau$, т.е.:

$$l(\tau; \delta)_u \in [0; d(l; \delta)_\tau] \text{ и } z(\tau; \delta)_u \in [0; d(z; \delta)_\tau]. \quad (4)$$

4. Из обработки исключаются позиции p_u ДСП пространства с координатами $u=1$ и $u=U.$
5. Двумерное структурное спектральное пространство трансформанты является перетрансформированным в соответствии с однонаправленностью градиента изменения значений элементов двухкомпонентных векторов, что задается выражением (2).

Координатный объект $\tilde{p}_u$ в соответствии с перетрансформацией обладает следующими свойствами:

– значения его координатных составляющих по осям L и Z имеют градиентную однонаправленность, а именно увеличение значения $l(\tau; \delta)_u$ длины цепочки незначимых компонент согласовывается с ростом величины значимой компоненты $z(\tau; \delta)_{U-u+1};$

– значения координатных составляющих в соответствии с особенностями формирования уплотненного ДСП пространства ограничены динамическим количеством допустимых значений, т.е.

$$l(\tau; \delta)_u \in [0; d(l; \delta)_\tau - 1]; \quad z(\tau; \delta)_{U-u+1} \in [1; d(z; \delta)_\tau - 1], \\ u = \overline{2, U-1}; \quad (5)$$

– значения координатных составляющих $l(\tau; \delta)_u$ и $z(\tau; \delta)_{U-u+1}$ отличаются характерной неравномер-

ностью распределения и нелинейностью изменений по осям L и $Z.$

Данные свойства создают условия для интерпретации координатных объектов $\tilde{p}_u$ ДСП пространства как двухэлементных биадических чисел в условиях градиентной однонаправленности. В связи с этим предлагается формировать идентификаторы $I(\tau; \delta)_u$ позиций $\tilde{p}_u$, как кодовые значения биадических чисел в условиях градиентной однонаправленности.

Тогда функциональное преобразование $F(I)$ для идентификации координатного объекта $\tilde{p}_u$, т.е.

$$F(I): \{l(\tau; \delta)_u; z(\tau; \delta)_{U-u+1}\} \rightarrow I(\tau; \delta)_u$$

примет следующий вид:

$$I(\tau; \delta)_u = (l(\tau; \delta)_u (d(z; \delta)_\tau - 1) + z(\tau; \delta)_{U-u+1}). \quad (6)$$

Данное выражение позволяет построить пространство $I(\tau; \delta)$ идентификаторов по дискретным позициям ДСП пространства в условиях градиентной однонаправленности. В результате формируется вектор $I(\tau; \delta)$ идентификаторов, т.е.
 $I(\tau; \delta) = \{I(\tau; \delta)_2; \dots I(\tau; \delta)_u; \dots I(\tau; \delta)_{U-1}\}.$

Для создания эффективного синтаксического представления последовательности $I(\tau; \delta)$ идентификаторов координатных объектов ДСП пространства предлагается вектор идентификаторов в условиях наложения ограничений на неравенство парных элементов интерпретировать как одномерные объектно-позиционные числа с основанием $Q(\tilde{p}_u; \delta)$, для элементов которого выполняется условие гибкого неравенства (с синдромом парного неравенства $S(\tau; \delta)$).

В этом случае вектор $I(\tau; \delta)$ идентификаторов без учета ограничений заменяется двумя векторами, а именно вектором $\hat{I}(\tau; \delta)$ идентификаторов в условиях наложения ограничений на неравенство парных элементов и соответствующим синдромом $S(\tau; \delta)$, т.е.:

$$I(\tau; \delta) \xrightarrow{\varphi_s} \{\hat{I}(\tau; \delta); S(\tau; \delta)\},$$

где φ_s – функционал выявления условия неравенства для смежных (парных) идентификаторов.

Поэтому эффективное синтаксическое кодирование вектора идентификатора ДСП пространства должно осуществляться с учетом дополнительного использования синдрома $S(\tau; \delta)$ объектно-позиционного числа в условиях наличия гибкого неравенства его парных элементов.

Для этого необходимо интегрировать в кодообразующую систему вектора $\hat{I}(\tau; \delta)$ идентификаторов

синдром $S(\tau; \delta)$ объектно-позиционного числа. Такое интегрирование предлагается проводить на основе следующих технологических аспектов:

1. Первый аспект заключается в учете гибкости неравенства парных идентификаторов. Для этого вводится корректирующая вспомогательная величина $\theta'(\tau; \delta)_u$, а именно:

$$\theta'(\tau; \delta)_u = \begin{cases} \hat{I}(\tau; \delta)_u, & \rightarrow \hat{I}(\tau; \delta)_u < \hat{I}(\tau; \delta)_{u-1} \vee s(\tau; \delta)_u = 0; \\ \hat{I}(\tau; \delta)_u - 1, & \rightarrow \hat{I}(\tau; \delta)_u > \hat{I}(\tau; \delta)_{u-1} \vee s(\tau; \delta)_u = 0; \\ \hat{I}(\tau; \delta)_u, & \rightarrow s(\tau; \delta)_u = 1. \end{cases}$$

2. Второй технологический аспект относится к формированию весовых коэффициентов $Q(s(\tau; \delta)_u)$ и описывается следующей системой выражений:

$$Q(s(\tau; \delta)_u) = \begin{cases} \hat{Q}(\tilde{p}_u; \delta) \rightarrow I(\tau; \delta)_u \neq I(\tau; \delta)_{u+1}, s(\tau; \delta)_u = 0; \\ \hat{Q}(\tilde{p}_u; \delta) \rightarrow I(\tau; \delta)_u = I(\tau; \delta)_{u+1}, s(\tau; \delta)_u = 1. \end{cases}$$

В этом случае последовательность $\hat{I}(\tau; \delta)$ идентификаторов можно трактовать как одномерное двухосновное $\{Q(\tilde{p}_u; \delta); \hat{Q}(\tilde{p}_u; \delta)\}$ объектно-позиционное число с учетом наличия гибкости условия неравенства смежных элементов, т.е. с учетом синдрома $S(\tau; \delta)$.

С учетом введенных технологических аспектов система соотношений для одномерного двухосновного объектно-позиционного кодирования в условиях гибкого неравенства парных элементов примет следующий вид:

$$\hat{E}(s; \tau; \delta) = \sum_{u=2}^{U-1} \theta'(\tau; \delta)_u Q(\tilde{p}_u; \delta)^{v_u} \cdot \hat{Q}(\tilde{p}_u; \delta)^{\hat{v}_u}. \quad (7)$$

Здесь $\hat{E}(s; \tau; \delta)$ – кодовое значение одномерного двухосновного объектно-позиционного числа в условиях гибкого неравенства парных элементов, т.е. с учетом синдрома $S(\tau; \delta)$; v_u и $\hat{v}_u$ – количество идентификаторов, оставшихся не обработанными на u -м шаге кодирования, и соответственно для которых выполняется условие равенства, т.е. $s(\tau; \delta)_u = 1$ и выполняется условие неравенства, т.е. $s(\tau; \delta)_u = 0$. Для u -го шага кодирования выполняется равенство: $v_u + \hat{v}_u = U - u - 1$.

$Q'(s; \tau; \delta)_{U-u-1}$ – весовой коэффициент вспомогательного элемента $\theta'(\tau; \delta)_u$ с дополнительным учетом информации о величине $s(\tau; \delta)_u$.

Таким образом, соотношение (7) образует технологическое ядро эффективного синтаксического представления сегмента видеокadra на основе формирования уплотненного двумерного спектрального структурного пространства с последующим одномерным двухосновным объектно-

позиционным кодированием в пространстве идентификаторов в условиях гибкого неравенства парных элементов.

5. Выводы

1. Показано, что существующие информационные технологии обработки и передачи видеопотока не обеспечивают компромисс между ростом информационной интенсивности динамического видеоресурса и ограниченностью пропускных способностей инфокоммуникационных технологий на базе бортовых беспилотных комплексов. Следовательно, существует дисбаланс между величиной информационной интенсивности кодированного видеопотока и целостностью его синтаксического и семантического содержания.

2. Разработаны методологические аспекты обработки динамического видеоресурса для повышения качества дистанционного видеосервиса, которые базируются на следующих технологических составляющих:

- 1) метод идентификации уплотненного двумерного спектрального структурного пространства по его дискретным позициям (координатным объектам) на основе биадического принципа в условиях градиентной однонаправленности с учетом наложения ограничений на динамическое количество значений, которые принимают элементы структурных составляющих ДСП пространства;
- 2) концепция внутрикадрового эффективного синтаксического кодирования сегментов видеокadров на основе их трансформирования. Отличительные особенности концепции состоят в том, что формируется уплотненное ДСП пространства по векторам структурных характеристик с последующей идентификацией их координатных объектов с сохранением целостности информации в условиях однонаправленности градиента;
- 3) технологическое ядро эффективного синтаксического представления сегмента видеокadra на основе: формирования уплотненного двумерного спектрального структурного пространства с последующим одномерным двухосновным объектно-позиционным кодированием в пространстве идентификаторов в условиях гибкого неравенства парных элементов; классификации пар идентификаторов уплотненного ДСП пространства по наличию условия неравенства и результату сравнения монотонности.

Литература: 1. *Кашкин В.Б.* Цифровая обработка аэрокосмических изображений: Конспект лекций / В.Б. Кашкин, А.И. Сухинин. Красноярск: ИПК СФУ, 2008. 121 с. 2. *Алімнієв А.М., Бараннік В.В., Белікова Т.В., Сідченко С.О.* Теоретичні основи створення технологій протидії прихованим інформаційним атакам в сучасній гібри-

- дній війні // Системи обробки інформації. 2017. № 4(150). С. 113-121. 3. Шульгин С.С., Красноруцкий А.А., Кулица О.С. Исследование характеристик сервиса дистанционного предоставления видеослужб при управлении в кризисных ситуациях // Открытые компьютерные информационные интегрированные технологии. 2015. № 70. С. 263-270. 4. Gonzalez R. Digital image processing. М.: Technosphere, 2005. 1072 p. 5. Миано Дж. Форматы и алгоритмы сжатия изображений в действии: учебное пособие. М.: Триумф, 2003. 336 с. 6. Лидовский В.В. Теория информации. М.: Компания Спутник+, 2004. 111 с. 7. Yudin O., Frolov O., Ziubina R. Quantitative quality indicators of the invariant spatial method of compressing video data, [Problems of Infocommunications Science and Technology (PIC S&T), Second International Scientific-Practical Conference]. P. 227-229, doi.org/10.1109/infocommst.2015.7357320. 8. Красильников Н.Н. Цифровая обработка изображений. Москва: Вузовская книга, 2011. 320 с. 9. Stankiewicz O., Wegner K., Karwowski D., Stankowski J., Klimaszewski K., Grajek T. Encoding mode selection in HEVC with the use of noise reduction, [International Conference on Systems, Signals and Image Processing (IWSSIP)], Poznan. 2017. P. 1-6. 10. Zhang Y., Negahdaripour S., Li Q. Error-resilient coding for underwater video transmission, [OCEANS 2016 MTS/IEEE Monterey], Monterey, CA. 2016. P. 1-7. 11. Wang S., Zhang X., Liu X., Zhang J., Ma S., Gao W. Utility-Driven Adaptive Preprocessing for Screen Content Video Compression, [Transactions on Multimedia]. Vol. 19. No. 3. 2017. P. 660-667. 12. Ватолін Д., Ратушняк А., Смирнов М., Юкин В. Методы сжатия данных. Устройство архиваторов, сжатие изображений и видео: учебное пособие. М.: ДИАЛОГ – МИФИ. 2003. 384с. 13. Barannik V.V., Ryabukha Yu.N., Podlesnyi S.A. Structural slotting with uniform redistribution for enhancing trustworthiness of information streams, [Telecommunications and Radio Engineering], English translation of Elektrosvyaz and Radiotekhnika, No. 76 (7), 2017. P. 607. doi: /10.1615/TelecomRadEng.v76.i7.40. 14. Alimpiev A.N., Barannik V.V., Sidchenko S.A. The method of cryptocompression presentation of videoinformation resources in a generalized structurally positioned space, [Telecommunications and Radio Engineering], English translation of Elektrosvyaz and Radiotekhnika, No. 76 (6), 2017. pp. 521-534, doi: 10.1615/TelecomRadEng.v76.i6.60. 15. Хаханов В.И., Побеженко И.А. Модели и архитектура вейвлет преобразований для стандарта JPEG 2000 // АСУ и приборы автоматики. 2007. № 2(139). С. 4-12. 16. Гуржий П.Н., Бойко Ю.П., Третьяк В.Ф. Адаптивне одноосновне позиційне кодування масивів довжин серій двійкових елементів // Радіоелектроніка та інформатика. 2013. № 2. С. 12-17. 17. Baranni, V., Krasnorutskiy A., Ryabukha Y.N., Okladnoy D.E. Model intelligent processing of aerial photographs with a dedicated key features interpretation, [Modern Problems of Radio Engineering], Telecommunications and Computer Science (TCSET), 2016, Lviv. 2016. pp. 736, doi: 10.1109/TCSET.2016.7452167. 18. Баранник В.В., Рябуха Ю.Н. Концептуальный метод повышения безопасности дистанционного видеонаблюдения ресурса в системе аэромониторинга кризисных ситуаций на основе интеллектуальной обработки видеок кадров // Радиоэлектронные компьютерные системы. 2015. № 3. С. 19-21. 19. Barannik V., Tarasenko D. Method of encoding a video frame based on the identification vectors seal structural transformed space // Radioelectronics & informatics. 2016. No. 4. P. 4-11.
- Transliterated bibliography:**
1. Kashkin V.B. Cifrovaya obrabotka aerokosmicheskikh izobrazenii. IPK SFU, Krasnoyarsk, 2008. 121 p.
 2. Alimpiev A.M., Barannik V.V., Belikova T.V., Sidchenko S.O. Teoretichni osnovi stvorenniya tehnologii protidii prihovanim informacii in atakam v suchasni gibridni viini // Information Processing Systems. No. 4(150). 2017. P. 113-121.
 3. Shulgin S.S., Krasnoruckii A.A., Kulica O.S. Issledovanie harakteristik servisa distancionnogo predostavlenniya videouslug pri upravlenii v krizisnih situaciyah // Open information and computer integrated technologies. 2015. No. 70. P. 263-270.
 4. Gonzalez R., Woods R. Digital Image Processing, Technosphere, Moscow, 2002. 1072 p.
 5. Miano J. Formatyi i algoritmyi szhatiya izobrazheniy v deystvii. Moscow: Triumf. 2003. 336 p.
 6. Lidovskii V.V. Teoriya informacii. Moscow: Kompany Sputnik+.2004. 111 p.
 7. Yudin O., Frolov O., Ziubina R. Quantitative quality indicators of the invariant spatial method of compressing video data, [Problems of Infocommunications Science and Technology (PIC S&T), Second International Scientific-Practical Conference]. P. 227-229, doi.org/10.1109/infocommst.2015.7357320.
 8. Krasilnikov N.N. Cifrovaya obrabotka izobrazenii. Moscow: The University book. 2011. 320 p.
 9. Stankiewicz O., Wegner K., Karwowski D., Stankowski J., Klimaszewski K., Grajek T. Encoding mode selection in HEVC with the use of noise reduction, [International Conference on Systems, Signals and Image Processing (IWSSIP)], Poznan, 2017. P. 1-6.
 10. Zhang Y., Negahdaripour S., Li Q. Error-resilient coding for underwater video transmission, [OCEANS 2016 MTS/IEEE Monterey], Monterey, CA. 2016. P. 1-7.
 11. Wang S., Zhang X., Liu X., Zhang J., Ma S., Gao W. Utility-Driven Adaptive Preprocessing for Screen Content Video Compression, [Transactions on Multimedia]. Vol. 19. 2017. No. 3. P. 660-667.
 12. Vatolin D., Ratushnyak A., Smirnov M., Yukin V. Metodi szhatiya dannih. Ustroistvo arhivatorov szatie izobrazeniii video: uchebnoe posobie. Moscow. DIALOG MIFI. 2003. 384 p.
 13. Barannik V.V., Ryabukha Yu.N., Podlesnyi S.A. Structural slotting with uniform redistribution for enhancing trustworthiness of information streams, [Telecommunications and Radio Engineering], English translation of Elektrosvyaz and Radiotekhnika. No. 76 (7). 2017. pp. 607, doi: /10.1615/TelecomRadEng.v76.i7.40.

14. Alimpiev A.N., Barannik V.V., Sidchenko S.A. The method of cryptocompression presentation of videoinformation resources in a generalized structurally positioned space, [Telecommunications and Radio Engineering], English translation of *Elektrosvyaz and Radiotekhnika*. No. 76 (6). 2017. pp. 521-534, doi: 10.1615/TelecomRadEng.v76.i6.60.
15. Hahanov V.I., Pobejenko I.A. Modeli i arhitektura veivlet preobrazovanii dlya standarta JPEG 2000 // *Control Systems and Automation Devices*. 2007. No. 2(139). pp. 4-12.
16. Gurjii P.N., Boiko Yu.P., Tretyak V.F. Adaptivne odnoosnovne poziciine koduvannya masiviv dovjin serii dviikovih elementiv // *RadioElectronics & Informatics Journal*. 2013. No 2. P. 12-17.
17. Baranni, V., Krasnorutskiy A., Ryabukha Y.N., Okladnoy D.E. Model intelligent processing of aerial photographs with a dedicated key features interpretation, [Modern Problems of Radio Engineering], *Telecommunications and Computer Science (TCSET)*, 2016. Lviv, 2016. P. 736, doi: 10.1109/TCSET.2016.7452167.
18. Barannik V.V., Ryabuha Yu.N. Konceptualnii metod povisheniya bezopasnosti distancionnogo videoi nfor-

macionnogo resursa v sisteme aeromonitoringa krizisnih situacii na osnove intellektualnoi obrabotki videokadrov // *Radioelectronic computer systems*. 2015. No. 3. P. 19-21.

19. Barannik V., Tarasenko D. Method of encoding a video frame based on the identification vectors seal structural transformed space // *Radioelectronics & informatics*. 2016. No. 4. P. 4-11.

Поступила в редколлегию 12.11.2017

Рецензент: д-р техн. наук, проф. Безрук В.М..

Тарасенко Дмитрий Анатолієвич, аспірант Черкаського державного технологічного університету. Научні інтереси: кодування і захист інформації для передачі в телекомунікаційних системах. Адрес: Україна, 61000, Харків, ул. Сумська, 77/79. E-mail: barannik_v_v@mail.ru.

Tarasenko Dmitry Anatolyevich, post-graduate student, Cherkasy State Technological University. Scientific interests: encoding and protection of information for transmission in telecommunication systems. Address: Ukraine, Kharkov, st. Sumska, 77/79. E-mail: barannik_v_v@mail.ru.

РОЗРОБКА ТА ДОСЛІДЖЕННЯ ЛІНІЙНОЇ ОПТИМІЗАЦІЙНОЇ МОДЕЛІ ШВИДКОЇ ПЕРЕМАРШРУТИЗАЦІЇ З БАЛАНСУВАННЯМ НАВАНТАЖЕННЯ В ТЕЛЕКОМУНІКАЦІЙНИХ МЕРЕЖАХ

ЛЕМЕШКО О.В., ЄРЕМЕНКО О.С.

Пропонується оптимізаційна модель швидкої перемаршрутизації з балансуванням навантаження в телекомунікаційних мережах, новизна якої полягає в тому, що оптимізаційну задачу балансування навантаження в ході швидкої перемаршрутизації за умови захисту пропускної здатності каналів зв'язку представлено в лінійній формі. Таке рішення на практиці зменшує обчислювальну складність визначення маршрутних змінних, відповідальних за формування основного та резервного шляхів, і забезпечує збалансовану завантаженість каналів зв'язку мережі, яка відповідає вимогам концепції Traffic Engineering. Модель забезпечує реалізацію схем захисту каналів, вузлів і пропускної здатності для швидкої перемаршрутизації з балансуванням навантаження в телекомунікаційних мережах. Аналіз запропонованої моделі підтверджує її адекватність та ефективність з точки зору отримання оптимальних рішень для забезпечення збалансованого завантаження каналів зв'язку мережі та реалізації необхідних схем захисту елементів мережі (каналу, вузла та пропускної здатності).

Ключові слова: маршрутизація; потокова модель; MPLS; трафік інжиніринг; швидка перемаршрутизація; основний шлях; резервний шлях; схеми захисту.

Key words: routing; flow-based model; MPLS; Traffic Engineering; Fast ReRoute; primary path; backup path; protection schemes.

1. Вступ

Сучасні телекомунікаційні мережі (ТКМ) є складними технічними системами, які функціонують в умовах постійного впливу різних внутрішніх і зовнішніх факторів, що, як правило, негативно позначаються на їх ефективності та рівні якості обслуговування (Quality of Service, QoS) в цілому. До основних з таких факторів належать відмови мережного обладнання, викликані низькою надійністю окремих пристроїв, їх неправильною експлуатацією або конфігуруванням, перевантаженням або компрометацією [1-4]. У зв'язку з цим як ніколи актуальною є задача, пов'язана з побудовою так званих відмовостійких мереж (resilient networks), здатних забезпечити високий рівень якості обслуговування в умовах ймовірних відмов мережного обладнання.

Питанням побудови та функціонування відмовостійких мереж присвячено багато наукових і прикладних розробок, що стосуються нарощування функціоналу та підвищення рівня адаптивності протоколів різних рівнів OSI [5-7]. Особливий інтерес викликають дослідження, спрямовані на підвищення відмовостійкості телекомунікаційних мереж засобами мережного рівня OSI і, зокрема, протоколами маршрутизації. Прикладами подібних рішень служать протоколи захисту шлюзу HSRP, VRRP, GLBP та CARP [8, 9], а також технологія швидкої перемаршрутизації (Fast ReRoute, FRR), що застосовується в IP/MPLS-мережах для захисту елементів транспортної

мережі – каналу, вузла, шляху та пропускної здатності мережі в цілому [10-14].

Перелічені протоколи забезпечують підвищення відмовостійкості мережі, припускаючи введення ресурсної надмірності, коли поряд з визначенням основного маршруту (OM) одночасно розраховується і резервний маршрут (PM) відповідно до реалізованої схеми захисту. У зв'язку з цим при вирішенні завдань FRR важливо забезпечити збалансоване використання доступного мережного, перш за все, каналного ресурсу, щоб захист елемента мережі не призвів до її перевантаження та істотного зниження рівня QoS. Проблематиці реалізації моделі швидкої перемаршрутизації з балансуванням навантаження в мережах MPLS (MPLS Traffic Engineering Fast ReRoute, MPLS TE FRR) присвячено багато наукових публікацій [15-24].

Забезпечення узгодженого вирішення завдань швидкої перемаршрутизації та балансування навантаження зазвичай призводить до підвищення обчислювальної складності та зниження масштабованості протокольних рішень. Відомо, що ефективність протокольного рішення багато в чому визначається адекватністю та якістю покладеної в його основу математичної моделі розрахунку. Як показав проведений аналіз [24], порядок FRR і TE визначається в ході розв'язання оптимізаційних задач різного рівня складності. При цьому реалізація схеми захисту пропускної здатності мережі, як правило, призводить до нелінійного формулювання оптимізаційної задачі та відповідного зростання обчислювальної складності одержуваних рішень [8, 9, 12-14]. В роботах [25, 26] зроблено спробу отримати розв'язання задачі TE FRR із захистом пропускної здатності на основі впровадження дворівневої ієрархії розрахунків. Проте оптимізаційні задачі, які розв'язуються на різних ієрархічних рівнях, вдалося сформулювати в лінійній формі лише для випадку реалізації одношляхової стратегії маршрутизації, що ефективно в разі формування мережного трафіка великою кількістю низькошвидкісних потоків.

Тому актуальним є науково-практичне завдання, пов'язане з розробкою ефективної лінійної оптимізаційної моделі швидкої перемаршрутизації з балансуванням навантаження в телекомунікаційних мережах при захисті каналу, вузла та пропускної здатності з підтримкою саме багатошляхової стратегії маршрутизації.

2. Потокова модель швидкої перемаршрутизації з балансуванням навантаження в ТКМ

Нехай структура ТКМ описується за допомогою графа

$G = (R, E)$, в якому $R = \{R_i; i = \overline{1, m}\}$ – це множина маршрутизаторів мережі, а $E = \{E_{i,j}; i, j = \overline{1, m}; i \neq j\}$ – множина каналів зв'язку (КЗ). Позначимо через

$R_i^* = \{R_j^* : E_{j,i} \neq 0; j = \overline{1, m}; i \neq j\}$ підмножину маршрутизаторів, які є суміжними для маршрутизатора R_i , тоді як кількість каналів зв'язку в мережі визначимо через $n = |E|$, а кожному з них поставимо у відповідність його пропускну здатність $\phi_{i,j}$.

В рамках даної моделі з кожним k -м одноадресним потоком пов'язано ряд функціональних характеристик: S_k – маршрутизатор-відправник; D_k – маршрутизатор-одержувач; λ^k – середня інтенсивність пакетів k -го потоку, яка вимірюється в пакетах за секунду (1/с). Нехай K – множина потоків пакетів, які передаються в мережі, тоді $k \in K$.

Результатом розв'язання задачі швидкої перемаршрутизації з балансуванням навантаження в ТКМ є розрахунок двох типів маршрутних змінних – $x_{i,j}^k$ та $\bar{x}_{i,j}^k$, кожна з яких характеризує частку інтенсивності k -го потоку в каналі зв'язку $E_{i,j} \in E$, що входить до основного або резервного шляху відповідно.

У випадку, коли в мережі використовується одношляхова маршрутизація, на маршрутні змінні двох типів накладаються обмеження виду

$$x_{i,j}^k \in \{0;1\} \text{ та } \bar{x}_{i,j}^k \in \{0;1\}, \quad (1)$$

а для багатошляхової стратегії маршрутизації –

$$0 \leq x_{i,j}^k \leq 1 \text{ та } 0 \leq \bar{x}_{i,j}^k \leq 1. \quad (2)$$

Для забезпечення зв'язності маршрутів, які розраховуються, вводяться умови збереження потоку окремо для маршрутних змінних основного шляху:

$$\begin{cases} \sum_{j: E_{i,j} \in E} x_{i,j}^k - \sum_{j: E_{j,i} \in E} x_{j,i}^k = 0; k \in K, R_i \neq S_k, D_k; \\ \sum_{j: E_{i,j} \in E} x_{i,j}^k - \sum_{j: E_{j,i} \in E} x_{j,i}^k = 1; k \in K, R_i = S_k; \\ \sum_{j: E_{i,j} \in E} x_{i,j}^k - \sum_{j: E_{j,i} \in E} x_{j,i}^k = -1; k \in K, R_i = D_k; \end{cases} \quad (3)$$

і окремо для маршрутних змінних резервного шляху [12-14]:

$$\begin{cases} \sum_{j: E_{i,j} \in E} \bar{x}_{i,j}^k - \sum_{j: E_{j,i} \in E} \bar{x}_{j,i}^k = 0; k \in K, R_i \neq S_k, D_k; \\ \sum_{j: E_{i,j} \in E} \bar{x}_{i,j}^k - \sum_{j: E_{j,i} \in E} \bar{x}_{j,i}^k = 1; k \in K, R_i = S_k; \\ \sum_{j: E_{i,j} \in E} \bar{x}_{i,j}^k - \sum_{j: E_{j,i} \in E} \bar{x}_{j,i}^k = -1; k \in K, R_i = D_k. \end{cases} \quad (4)$$

3. Умови захисту каналу, вузла та пропускну здатності при швидкій перемаршрутизації з балансуванням навантаження в ТКМ

Як показав аналіз [7, 12-14], в ході швидкої перемаршрутизації можуть підтримуватися кілька основних схем захисту елементів мережі: вузла, каналу, шляху та його пропускну здатності. У роботах [12, 13] отримано умови в аналітичному вигляді для підтримки зазначених схем захисту як складових відповідних математичних моделей.

В роботі [13] пропонується при реалізації схеми захисту каналу $E_{i,j} \in E$ на маршрутні змінні $\bar{x}_{i,j}^k$, що відповідають за визначення резервного шляху, накласти додаткові обмеження, аналогічні (1). При цьому в разі реалізації одношляхової стратегії маршрутизації має місце таке обмеження:

$$\bar{x}_{i,j}^k \in \{0; \delta_{i,j}^k\}, \quad (5)$$

тоді як при багатошляховій маршрутизації

$$0 \leq \bar{x}_{i,j}^k \leq \delta_{i,j}^k, \quad (6)$$

де

$$\delta_{i,j}^k = \begin{cases} 0, & \text{при захисті каналу зв'язку } E_{i,j}; \\ 1, & \text{в іншому випадку.} \end{cases} \quad (7)$$

Виконання умов (5)-(7) гарантує, що канал $E_{i,j} \in E$, який захищається, не буде використовуватися резервним маршрутом при одношляховій маршрутизації. Умови (5)-(7) носять лінійний характер на відміну від нелінійних рішень, запропонованих в роботі [7], що сприяє зниженню обчислювальної складності отримання кінцевих протокольних рішень.

При реалізації схеми захисту вузла $R_i \in R$ умови (5)-(7) узагальнюються на випадок захисту множини каналів зв'язку, інцидентних вузлу, що захищається [12, 13]. Тоді в разі використання одношляхової стратегії вводиться система умов:

$$\bar{x}_{i,j}^k \in \{0; \delta_{i,j}^k\} \text{ при } R_j \in R_i^*, j = \overline{1, m}, \quad (8)$$

а для багатошляхового випадку –

$$0 \leq \bar{x}_{i,j}^k \leq \delta_{i,j}^k \text{ при } R_j \in R_i^*, j = \overline{1, m}, \quad (9)$$

де вибір значень $\delta_{i,j}^k$ підпорядковується умові (7).

Таким чином, виконання вимог умов (8) та (9) гарантує захист вузла $R_i \in R$, забороняючи використання резервним маршрутом всіх каналів, які виходять з даного вузла. Оскільки захисту підлягають лише транзитні маршрутизатори, то заборона на використання вихідних каналів відповідно до умов (4) запобігає включенню до резервного шляху і вхідних каналів для даного вузла R_i , що в результаті сприяє захисту вузла в цілому. Варто зазначити, що умови захисту заздалегідь визначених вузлів і

каналів мережі, як правило, є лінійними, а їх врахування критично не позначається на складності обчислення маршрутних змінних $x_{i,j}^k$ та $\bar{x}_{i,j}^k$, що відповідають за формування множини основних і резервних маршрутів.

Умови захисту пропускної здатності мережі, які представляються через умови запобігання перевантаження каналів зв'язку в ході реалізації швидкої перемаршрутизації, коли деякі потоки можуть перемикатися на резервні маршрути, мають вигляд:

$$\sum_{k \in K} \lambda^k \cdot \max[x_{i,j}^k, \bar{x}_{i,j}^k] \leq \phi_{i,j}, \quad E_{i,j} \in E. \quad (10)$$

Умови (10), представлені в [9], запропоновано використовувати в такому вигляді:

$$\frac{1}{2} \sum_{k \in K} \lambda^k [x_{i,j}^k + \bar{x}_{i,j}^k + |x_{i,j}^k - \bar{x}_{i,j}^k|] \leq \phi_{i,j}, \quad E_{i,j} \in E. \quad (11)$$

Проте умови захисту пропускної здатності мережі (10) та (11) є нелінійними, що негативно позначається на обчислювальній складності відповідних протокольних рішень. У роботі [25, 26] завдяки введенню дворівневої ієрархії розрахунків відповідно до принципу прогнозування взаємодій теорії ієрархічних багаторівневих систем вдалось ці умови отримати в лінійній формі, але тільки для випадку реалізації одношляхової маршрутизації. Тому в даній роботі для забезпечення лінійного вигляду умов захисту пропускної здатності мережі при реалізації як одношляхової, так і багатошляхової маршрутизації пропонується ввести такі модифіковані умови запобігання перевантаження з метою забезпечення балансування навантаження в мережі:

$$\sum_{k \in K} \lambda^k \cdot u_{i,j}^k \leq \alpha \cdot \phi_{i,j}, \quad E_{i,j} \in E \quad (12)$$

при

$$x_{i,j}^k \leq u_{i,j}^k \text{ та } \bar{x}_{i,j}^k \leq u_{i,j}^k, \quad (13)$$

де $u_{i,j}^k$ також є керуючими змінними

$$0 \leq u_{i,j}^k \leq 1 \quad (14)$$

і являють собою верхній поріг (ВП) значень маршрутних змінних основних і резервних шляхів, тоді як α представляє собою додаткову керуючу змінну, яка чисельно визначає верхній поріг завантаженості каналів зв'язку мережі та відповідає таким умовам [17-19]:

$$0 \leq \alpha \leq 1. \quad (15)$$

Критерієм оптимальності рішень задач MPLS TE FRR за аналогією з результатами, отриманими в роботах [17-19], буде мінімум порогу α , введеного в (12), тобто

$$\min_{x, \bar{x}, \alpha} \alpha. \quad (16)$$

Таким чином, рішення вихідної технологічної задачі швидкої перемаршрутизації з балансуванням навантаження в телекомунікаційних мережах із захистом каналу, вузла та пропускної здатності було зведено до

розв'язання оптимізаційної задачі лінійного програмування з критерієм (16) при наявності обмежень (1)-(6), (8), (9), (12)-(15).

4. Дослідження запропонованої моделі

Аналіз запропонованої моделі швидкої перемаршрутизації з балансуванням навантаження в ТКМ проведено на множині мережних конфігурацій для різного числа потоків та їх характеристик. Особливості роботи моделі TE FRR продемонструємо на розрахунковому прикладі. При цьому структура досліджуваної мережі показана на рис. 1, а в розривах каналів зв'язку мережі вказана їх пропускна здатність.

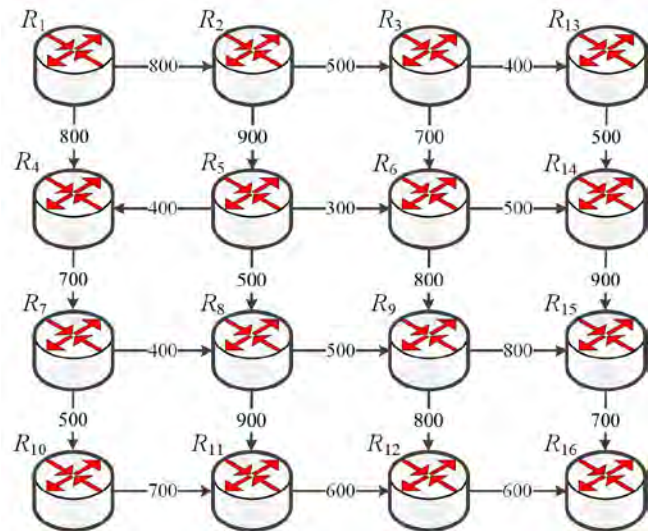


Рис. 1. Структура досліджуваної мережі

Нехай в мережі необхідно забезпечити розв'язання задачі швидкої перемаршрутизації двох потоків. При цьому пакети першого потоку передавались від вузла R_1 до вузла R_{16} . Пакети другого потоку – від R_5 до R_{12} . Припустимо, що інтенсивність цих потоків зміню-

валась в таких межах: $\lambda^1 = 10 \div 400$ 1/с та $\lambda^2 = 10 \div 400$ 1/с. Розглянемо, як веде себе верхній поріг завантаженості каналів зв'язку мережі (15) залежно від реалізованої схеми захисту каналу, вузла та пропускної здатності. В табл. 1 показані мінімальні та максимальні значення виграшу щодо значень критерію (16) при реалізації багатошляхової маршрутизації у порівнянні з використанням одношляхової маршрутизації при захисті кожного з каналів зв'язку мережі окремо. Таким чином, при захисті каналів зв'язку використання моделі (1)-(16) дозволяє покращити критерій (16) в середньому від 37,12 до 59,41%. Для наочності на рис. 2 показано залежність верхнього порогу завантаженості каналів зв'язку від значень інтенсивностей потоків, якщо реалізується, наприклад, схема захисту каналу $E_{8,11}$ при багатошляховій (рис. 2, а) або одношляховій маршрутизації (рис. 2, б).

Таблиця 1

Канал зв'язку, що захищається	Виграш, %	
	min	max
$E_{1,2}$	28,57	58,33
$E_{2,3}$	28,57	61,54
$E_{1,4}$	37,5	58,33
$E_{2,5}$	47,37	58,33
$E_{3,6}$	44,44	61,54
$E_{5,4}$	37,5	61,54
$E_{5,6}$	37,5	61,54
$E_{4,7}$	23,08	37,05
$E_{5,8}$	40,17	61,54
$E_{6,9}$	44,44	61,54
$E_{7,8}$	44,44	61,54
$E_{8,9}$	28,57	61,54
$E_{7,10}$	41,18	58,33
$E_{8,11}$	44,44	61,54
$E_{9,12}$	16,67	60,55
$E_{10,11}$	41,18	58,33
$E_{11,12}$	23,08	58,33
$E_{3,13}$	47,37	61,54
$E_{13,14}$	47,37	61,54
$E_{6,14}$	47,37	61,54
$E_{14,15}$	47,37	61,54
$E_{9,15}$	47,37	61,54
$E_{15,16}$	16,67	58,33
$E_{12,16}$	28,57	58,33

Як показано на рис. 3, реалізація багатошляхової маршрутизації при TE FRR і захисті каналу $E_{8,11}$ дозволяє покращити значення критерію (16) від 44,44 до 61,54% у порівнянні з використанням одношляхової маршрутизації.

Більш докладно розглянемо випадок використання запропонованої моделі TE FRR при передачі пакетів двох потоків з інтенсивностями $\lambda^1 = 400$ 1/с та $\lambda^2 = 300$ 1/с (табл. 1). При цьому завантаженість для кожного каналу зв'язку $E_{i,j} \in E$ визначалася як

$$\alpha_{i,j} = \frac{\sum_{k \in K} u_{i,j}^k \lambda^k}{\varphi_{i,j}}. \quad (17)$$

У табл. 2 також показано порядок багатошляхової маршрутизації та балансування двох потоків за каналами зв'язку мережі з використанням запропонованої моделі TE FRR (1)-(16). При цьому α (15) є максимальним значенням серед множини коефіцієнтів $\alpha_{i,j}$ (17). Відповідно до отриманих результатів розрахунку (табл. 2), реалізація одношляхової маршрутизації при захисті каналу $E_{8,11}$ забезпечила значення $\alpha = 0.8$, тоді як використання багатошляхової маршрутизації – $\alpha = 0.39$, що на 51,25% краще, ніж при одношляховій стратегії.

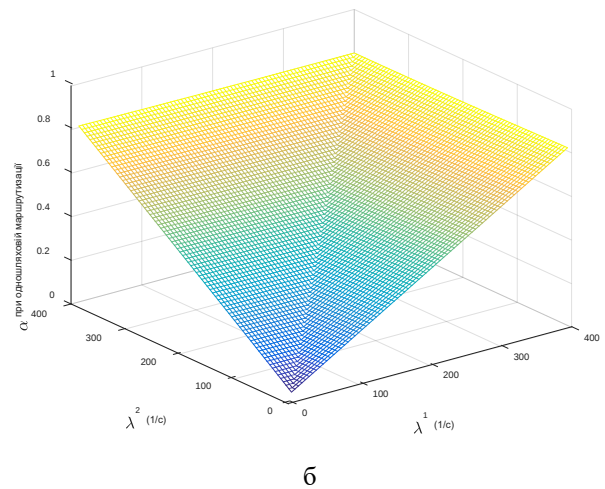
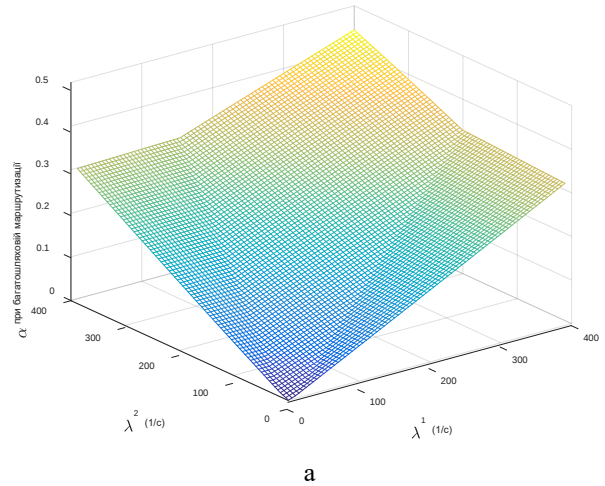


Рис. 2. Залежність верхнього порогу завантаженості каналів зв'язку мережі від значень інтенсивностей потоків, якщо реалізується схема захисту каналу $E_{8,11}$: а – при багатошляховій маршрутизації; б – при одношляховій маршрутизації

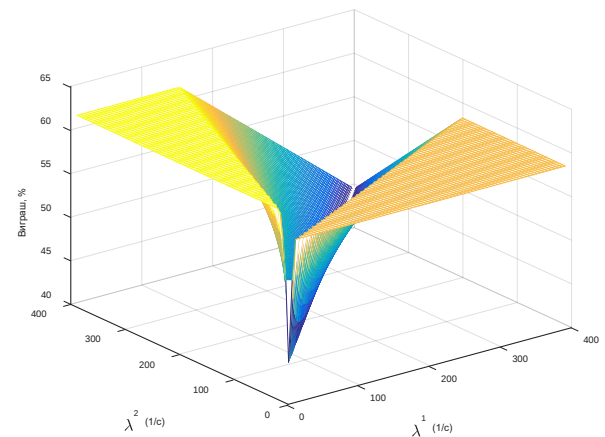


Рис. 3. Виграш за критерієм (16) від реалізації багатошляхової маршрутизації у порівнянні з використанням одношляхової маршрутизації (захист каналу $E_{8,11}$)

Таблиця 2

КЗ	Багатошляхова маршрутизація						$\alpha_{i,j}$	Одношляхова маршрутизація						$\alpha_{i,j}$
	Перший потік			Другий потік				Перший потік			Другий потік			
	ОМ	РМ	ВП	ОМ	РМ	ВП		ОМ	РМ	ВП	ОМ	РМ	ВП	
$E_{1,2}$	257,50	256,93	267,91	0	0	0	0,37	400	400	400	0	0	0	0,50
$E_{2,3}$	188,42	194,44	194,44	0	0	0	0,39	0	0	0	0	0	0	0
$E_{1,4}$	142,50	143,07	192,67	0	0	0	0,35	0	0	0	0	0	0	0
$E_{2,5}$	69,08	62,49	142,64	0	0	0	0,32	400	400	400	0	0	0	0,44
$E_{3,6}$	133,28	138,09	171,75	0	0	0	0,36	0	0	0	0	0	0	0
$E_{5,4}$	8,62	9,08	32,03	83,38	92,69	106,81	0,35	0	0	0	300	300	300	0,75
$E_{5,6}$	19,99	23,58	23,57	89,15	93,09	93,09	0,39	0	0	0	0	0	0	0
$E_{4,7}$	151,12	152,15	163,14	83,38	92,69	99,06	0,37	0	0	0	300	300	300	0,43
$E_{5,8}$	40,47	29,83	50,14	127,47	114,22	134,53	0,37	400	400	400	0	0	0	0,80
$E_{6,9}$	95,78	102,81	153,04	89,15	93,09	146,44	0,37	0	0	0	0	0	0	0
$E_{7,8}$	68,42	43,88	82,50	32,66	6,52	52,99	0,34	0	0	0	0	0	0	0
$E_{8,9}$	69,45	73,71	73,71	114,23	120,74	120,74	0,39	400	400	400	0	0	0	0,80
$E_{7,10}$	82,70	108,27	108,27	50,72	86,17	86,17	0,39	0	0	0	300	300	300	0,60
$E_{8,11}$	39,44	0	137,71	45,90	0	137,98	0,31	0	0	0	0	0	0	0
$E_{9,12}$	49,00	60,85	76,63	203,38	213,83	222,45	0,37	400	400	400	0	0	0	0,50
$E_{10,11}$	82,70	108,27	140,51	50,72	86,17	111,48	0,36	0	0	0	300	300	300	0,43
$E_{11,12}$	122,14	108,27	126,88	96,62	86,17	100,64	0,382	0	0	0	300	300	300	0,50
$E_{3,13}$	55,14	56,35	78,87	0	0	0	0,34	0	0	0	0	0	0	0
$E_{13,14}$	55,14	56,35	89,92	0	0	0	0,33	0	0	0	0	0	0	0
$E_{6,14}$	57,49	58,86	90,32	0	0	0	0,33	0	0	0	0	0	0	0
$E_{14,15}$	112,63	115,21	168,42	0	0	0	0,33	0	0	0	0	0	0	0
$E_{9,15}$	116,23	115,67	175,17	0	0	0	0,34	0	0	0	0	0	0	0
$E_{15,16}$	228,86	230,88	240,10	0	0	0	0,37	0	0	0	0	0	0	0
$E_{12,16}$	171,14	169,12	183,21	0	0	0	0,37	400	400	400	0	0	0	0,67

В табл. 3 показані мінімальні та максимальні значення виграшу щодо значень критерію (16) при реалізації багатошляхової маршрутизації у порівнянні з використанням одношляхової маршрутизації при захисті кожного з вузлів мережі окремо. Таким чином, використання моделі (1)-(16) при захисті вузлів мережі дозволяє покращити критерій (16) в середньому від 31,5 до 56,3 %.

Таблиця 3

Вузол, що захищається	Виграш, %	
	min	max
R_2	28,57	58,33
R_3	28,57	61,54
R_4	23,08	37,5
R_6	33,33	61,54
R_7	23,08	37,5
R_8	30,97	60,55
R_9	16,67	60,55
R_{10}	41,18	58,33
R_{11}	41,18	58,33
R_{13}	47,37	61,54
R_{14}	47,37	61,54
R_{15}	16,67	58,33

На рис. 4, наприклад, показано, що реалізація багатошляхової маршрутизації при захисті вузла R_9 дозволяє від 16,67 до 60,55% покращити значення критерію (16) у порівнянні з одношляховою маршрутизацією.

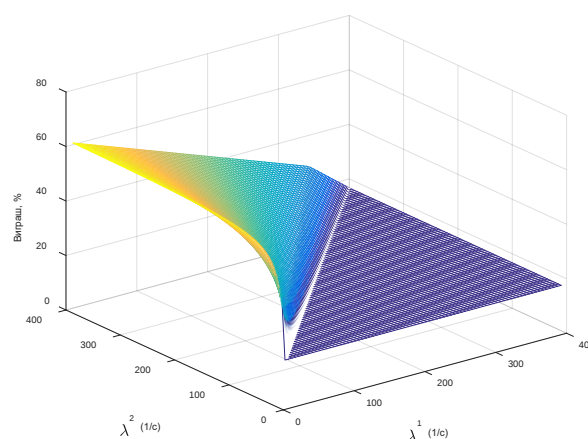


Рис. 4. Виграш за критерієм (16) від реалізації багатошляхової маршрутизації у порівнянні з використанням одношляхової маршрутизації (захист вузла R_9)

5. Висновки

Запропоновано математичну модель швидкої перемаршрутизації з балансуванням навантаження в телекомунікаційних мережах, представлену виразами (1)-(9) і (12)-(16). Новизною запропонованої моделі є те, що узгоджене рішення задач по ТЕ та FRR із захистом каналу, вузла та пропускної здатності забезпечується в ході рішення задачі лінійної оптимізації. Як критерій оптимальності виступав мінімум верхнього порогу завантаженості каналів зв'язку мережі (15), (16) потоками, що протікають як за основними, так і за резервними маршрутами. Перехід від нелінійних умов захисту пропускної здатності (10), (11) до лінійного аналогу (12) досягнуто шляхом деякого розширення числа змінних, що розраховуються, (13) і (14), які визначають верхній поріг для маршрутних змінних основного та резервного шляхів. Подібний підхід орієнтує на зниження обчислювальної складності при розрахунку маршрутних змінних, відповідальних за формування основного та резервного шляхів, і забезпечує збалансовану завантаженість каналів зв'язку мережі відповідно до вимог концепції Traffic Engineering.

Результати проведеного аналізу запропонованої моделі на ряді числових прикладів підтвердили її адекватність і можливість отримання оптимальних рішень задачі швидкої перемаршрутизації з балансуванням навантаження в телекомунікаційних мережах при реалізації різних схем захисту елементів мережі (каналу, вузла) та пропускної здатності. В роботі показано, що виграш від реалізації багатошляхової стратегії маршрутизації при ТЕ FRR дозволив знизити верхній поріг завантаженості каналів зв'язку в середньому від 37,12 до 59,41% при захисті каналу та від 31,5 до 56,3% при захисті вузла, що позитивно позначається і на рівні якості обслуговування в мережі в цілому.

Литература: 1. *White, R.* Computer Networking Problems and Solutions: An innovative approach to building resilient, modern networks. 1st Edition. / R. White, E. Banks. Addison-Wesley Professional, 2018. 832 p. 2. *White, R.* Navigating Network Complexity: Next-generation routing with SDN, service virtualization, and service chaining. / R. White, J.E. Tantsura. Addison-Wesley Professional, 2015. 320 p. 3. *Monge, A.S.* MPLS in the SDN Era: Interoperable Scenarios to Make Networks Scale to New Services. / A.S. Monge, K.G. Szarkowicz. O'Reilly Media, 2016. 920 p. 4. *Stallings, W.* Foundations of modern networking: SDN, NFV, QoE, IoT, and Cloud. / W. Stallings. Addison-Wesley Professional, 2015. 544 p. 5. *Alashaikh, A.* Exploring the logical layer to support differentiated resilience classes in multilayer networks / A. Alashaikh, D. Tipper, T. Gomes // Annals of Telecommunications. 2017. P. 1-17. 6. *Rak, J.* Information-driven network resilience: Research challenges and perspectives / J. Rak, D. Papadimitriou, H. Niedermayer, P. Romero // Optical Switching and Networking. January 2017. Vol. 23, part 2. P. 156-178. 7. *Lemeshko, O.* Effective solution for scalability and productivity improvement in fault-tolerant routing / O. Lemeshko, K. Arous, N. Tariki // IEEE 2015 Second International Scientific-Practical Conference Problems of Infocommunications Science and Technology (PIC S&T), 2015. P.

76-78. 8. *Lemeshko, O.V.* Fault-Tolerance Improvement for Core and Edge of IP Network / O.V. Lemeshko, O.S. Yermenko, N. Tariki, A.M. Hailan // IEEE 2016 XIth International Scientific and Technical Conference "Computer Sciences and Information Technologies" (CSIT), 2016. P. 161-164. 9. *Lemeshko, O.* Solution for the Default Gateway Protection within Fault-Tolerant Routing in an IP Network / O. Lemeshko, O. Yermenko, N. Tariki // International Journal of Electrical and Computer Engineering Systems. 2017. Volume 8, Number 1. P. 19-26. 10. *Papán, J.* The Survey of Current IPFRR Mechanisms / J. Papán, P. Segeč, P. Palúch, L. Mikuš, M. Moravčík // 2015 Federated Conference on Software Development and Object Technologies (SDOT 2015). Advances in Intelligent Systems and Computing. 2015. Vol. 511. P. 229-240. 11. *Nagy, M.* Optimization methods for improving IP-level fast protection for local shared risk groups with Loop-Free Alternates / M. Nagy, J. Tapolcai, G. Rétvári // Telecommunication Systems. May 2014. Vol. 56, Iss. 1. P. 103-119. 12. *Lemeshko, A.V.* Improvement of flow-oriented fast reroute model based on scalable protection solutions for telecommunication network elements / A.V. Lemeshko, O.S. Yermenko, N. Tariki // Telecommunications and Radio Engineering. 2017. Vol. 76, Issue 6. P. 477-490. 13. *Yermenko, O.S.* Fast ReRoute Scalable Solution with Protection Schemes of Network Elements / O.S. Yermenko, O.V. Lemeshko, N. Tariki // 2017 IEEE First Ukraine Conference on ELECTRICAL AND COMPUTER ENGINEERING (UKRCON). 2017. P. 783-788. 14. *Yermenko, O.* Research of Optimization model of Fault-Tolerant Routing with Bilinear Path Protection Criterion / O. Yermenko, O. Lemeshko, N. Tariki, A.M. Hailan // 2017 IEEE 2nd International Conference on Advanced Information and Communication Technologies (AICT). 2017. P. 219-222. 15. *Lin, S.C.* Control traffic balancing in software defined networks / S.C. Lin, P. Wang, M. Luo // Computer Networks. September 2016. Vol. 106. P. 260-271. 16. *RFC 4090* Fast reroute extensions to RSVP-TE for LSP tunnels / P. Pan, G. Swallow, A. Atlas // Internet Engineering Task Force (IETF), May 2005. 38 p. 17. *Seok, Y.* Dynamic Constrained Traffic Engineering for Multicast Routing / Y. Seok, Y. Lee, Y. Choi, C. Kim // Information Networking: Wired Communications and Management. September 2002. Vol. 2343. P. 278-288. 18. *Wang, Y.* Explicit routing algorithms for Internet Traffic Engineering / Y. Wang, Z. Wang // Eight International Conference on Computer Communications and Networks (Cat. No.99EX370). 11-13 Oct. 1999. P. 582-588. 19. *Seok, Y.* Dynamic Constrained Multipath Routing for MPLS Networks," / Y. Seok, Y. Lee, C. Kim, Y. Choi // in Proceedings Tenth International Conference on Computer Communications and Networks (Cat. No.01EX495), 15-17 Oct. 2001, pp. 348-353. 20. *RFC 6805* The Application of the Path Computation Element Architecture to the Determination of a Sequence of Domains in MPLS and GMPLS / D. King, A. Farrel // Internet Engineering Task Force (IETF), November 2012. 33 p. 21. *Paolucci, F.* A survey on the path computation element (PCE) architecture / F. Paolucci, F. Cugini, A. Giorgetti, N. Sambo, P. Castoldi // IEEE Communications Surveys & Tutorials. Fourth Quarter 2013. Vol. 15, Iss. 4. P. 1819-1841. 22. *Mendiola, A.* A survey on the contributions of Software-Defined Networking to Traffic Engineering / A. Mendiola, J. Astorga, E. Jacob, M. Higuero // IEEE Communications Surveys & Tutorials. Second quarter 2017. Vol. 19, Iss. 2. P. 918-953. 23. *Prabhavat, S.* On load distribution over multipath networks / S. Prabhavat, H. Nishiyama, N. Ansari, N. Kato // IEEE Communications Surveys & Tutorials. Third Quarter 2012. Vol. 14, Iss. 3. P. 662-680. 24. *Wang, N.* An overview of routing optimization for internet traffic engineering / N. Wang, K. Ho, G. Pavlou, M. Howarth // IEEE Communications Surveys & Tutorials. First Quarter 2008. Vol. 10, Iss. 1. P. 36-56. 25. *Lemeshko, O.* Two-level Method of Fast ReRouting in Software-Defined Net-

works / O. Lemeshko, O. Yeremenko, A.M. Hailan // 2017 Fourth International Scientific-Practical Conference Problems of Infocommunications Science and Technology (PIC S&T). 2017. P. 376-379. 26. *Lemeshko, O.* Enhanced method of fast re-routing with load balancing in software-defined networks / O. Lemeshko, O. Yeremenko // Journal of ELECTRICAL ENGINEERING. 2017. Vol. 68, Iss. 6. P. 444–454.

Транслітерований список літератури:

1. *White, R.* Computer Networking Problems and Solutions: An innovative approach to building resilient, modern networks. 1st Edition. / R. White, E. Banks. Addison-Wesley Professional, 2018. 832 p.
2. *White, R.* Navigating Network Complexity: Next-generation routing with SDN, service virtualization, and service chaining. / R. White, J.E. Tantsura. Addison-Wesley Professional, 2015. 320 p.
3. *Monge, A.S.* MPLS in the SDN Era: Interoperable Scenarios to Make Networks Scale to New Services. / A.S. Monge, K.G. Szarkowicz. O'Reilly Media, 2016. 920 p.
4. *Stallings, W.* Foundations of modern networking: SDN, NFV, QoE, IoT, and Cloud. / W. Stallings. Addison-Wesley Professional, 2015. 544 p.
5. *Alashaikh, A.* Exploring the logical layer to support differentiated resilience classes in multilayer networks / A. Alashaikh, D. Tipper, T. Gomes // Annals of Telecommunications. 2017. P. 1-17.
6. *Rak, J.* Information-driven network resilience: Research challenges and perspectives / J. Rak, D. Papadimitriou, H. Niedermayer, P. Romero // Optical Switching and Networking. January 2017. Vol. 23, part 2. P. 156-178.
7. *Lemeshko, O.* Effective solution for scalability and productivity improvement in fault-tolerant routing / O. Lemeshko, K. Arous, N. Tariki // IEEE 2015 Second International Scientific-Practical Conference Problems of Infocommunications Science and Technology (PIC S&T), 2015. P. 76-78.
8. *Lemeshko, O.V.* Fault-Tolerance Improvement for Core and Edge of IP Network / O.V. Lemeshko, O.S. Yeremenko, N. Tariki, A.M. Hailan // IEEE 2016 XIth International Scientific and Technical Conference "Computer Sciences and Information Technologies" (CSIT), 2016. P. 161-164.
9. *Lemeshko, O.* Solution for the Default Gateway Protection within Fault-Tolerant Routing in an IP Network / O. Lemeshko, O. Yeremenko, N. Tariki // International Journal of Electrical and Computer Engineering Systems. 2017. Volume 8, Number 1. P. 19-26.
10. *Papán, J.* The Survey of Current IPFRR Mechanisms / J. Papán, P. Segeč, P. Palúch, L. Mikuš, M. Moravčík // 2015 Federated Conference on Software Development and Object Technologies (SDOT 2015). Advances in Intelligent Systems and Computing. 2015. Vol. 511. P. 229-240.
11. *Nagy, M.* Optimization methods for improving IP-level fast protection for local shared risk groups with Loop-Free Alternates / M. Nagy, J. Topolcai, G. Rétvári // Telecommunication Systems. May 2014. Vol. 56, Iss. 1. P. 103-119.
12. *Lemeshko, A.V.* Improvement of flow-oriented fast reroute model based on scalable protection solutions for telecommunication network elements / A.V. Lemeshko, O.S. Yeremenko, N. Tariki // Telecommunications and Radio Engineering. 2017. Vol. 76, Issue 6. P. 477-490. DOI: 10.1615/TelecomRadEng.v76.i6.30

13. *Yeremenko, O.S.* Fast ReRoute Scalable Solution with Protection Schemes of Network Elements / O.S. Yeremenko, O.V. Lemeshko, N. Tariki // 2017 IEEE First Ukraine Conference on ELECTRICAL AND COMPUTER ENGINEERING (UKRCON). 2017. P. 783-788.

14. *Yeremenko, O.* Research of Optimization model of Fault-Tolerant Routing with Bilinear Path Protection Criterion / O. Yeremenko, O. Lemeshko, N. Tariki, A.M. Hailan // 2017 IEEE 2nd International Conference on Advanced Information and Communication Technologies (AICT). 2017. P. 219–222.

15. *Lin, S.C.* Control traffic balancing in software defined networks / S.C. Lin, P. Wang, M. Luo // Computer Networks. September 2016. Vol. 106. P. 260-271.

16. *RFC 4090* Fast reroute extensions to RSVP-TE for LSP tunnels / P. Pan, G. Swallow, A. Atlas // Internet Engineering Task Force (IETF), May 2005. 38 p.

17. *Seok, Y.* Dynamic Constrained Traffic Engineering for Multicast Routing / Y. Seok, Y. Lee, Y. Choi, C. Kim // Information Networking: Wired Communications and Management. September 2002. Vol. 2343. P. 278-288.

18. *Wang, Y.* Explicit routing algorithms for Internet Traffic Engineering / Y. Wang, Z. Wang // Eight International Conference on Computer Communications and Networks (Cat. No.99EX370). 11-13 Oct. 1999. P. 582-588.

19. *Seok, Y.* Dynamic Constrained Multipath Routing for MPLS Networks," / Y. Seok, Y. Lee, C. Kim, Y. Choi // in Proceedings Tenth International Conference on Computer Communications and Networks (Cat. No.01EX495), 15-17 Oct. 2001, pp. 348-353.

20. *RFC 6805* The Application of the Path Computation Element Architecture to the Determination of a Sequence of Domains in MPLS and GMPLS / D. King, A. Farrel // Internet Engineering Task Force (IETF), November 2012. 33 p.

21. *Paolucci, F.* A survey on the path computation element (PCE) architecture / F. Paolucci, F. Cugini, A. Giorgetti, N. Sambo, P. Castoldi // IEEE Communications Surveys & Tutorials. Fourth Quarter 2013. Vol. 15, Iss. 4. P. 1819-1841.

22. *Mendiola, A.* A survey on the contributions of Software-Defined Networking to Traffic Engineering / A. Mendiola, J. Astorga, E. Jacob, M. Higuero // IEEE Communications Surveys & Tutorials. Second quarter 2017. Vol. 19, Iss. 2. P. 918-953.

23. *Prabhavat, S.* On load distribution over multipath networks / S. Prabhavat, H. Nishiyama, N. Ansari, N. Kato // IEEE Communications Surveys & Tutorials. Third Quarter 2012. Vol. 14, Iss. 3. P. 662-680.

24. *Wang, N.* An overview of routing optimization for internet traffic engineering / N. Wang, K. Ho, G. Pavlou, M. Howarth // IEEE Communications Surveys & Tutorials. First Quarter 2008. Vol. 10, Iss. 1. P. 36-56.

25. *Lemeshko, O.* Two-level Method of Fast ReRouting in Software-Defined Networks / O. Lemeshko, O. Yeremenko, A.M. Hailan // 2017 Fourth International Scientific-Practical Conference Problems of Infocommunications Science and Technology (PIC S&T). 2017. P. 376-379.

26. *Lemeshko, O.* Enhanced method of fast re-routing with load balancing in software-defined networks / O. Lemeshko, O. Yeremenko // Journal of ELECTRICAL ENGINEERING. 2017. Vol. 68, Iss. 6. P. 444–454.

Надійшла до редколегії 11.12.2017

Рецензент: д-р техн. наук, проф. Безрук В.М.

Лемешко Олександр Віталійович, д-р техн. наук, професор кафедри інфокомунікаційної інженерії ХНУРЕ. Наукові інтереси: управління трафіком, оптимізація в телекомунікаціях, маршрутизація, трафік інжиніринг, QoS, QoE, ієрархічна маршрутизація, відмовостійка маршрутизація. Адреса: Україна, 61166, Харків, пр. Науки, 14, тел. +380577021320, e-mail: oleksandr.lemeshko.ua@ieee.org.

Єременко Олександра Сергіївна, канд. техн. наук., доцент, с.н.с., докторант кафедри інфокомунікаційної інженерії ХНУРЕ. Наукові інтереси: NGN, Future Internet, якість обслуговування, управління трафіком, відмовостійка маршрутизація, мережна безпека. Адреса: Україна, 61166, Харків, пр. Науки, 14, тел. +380577021320, e-mail: oleksandra.yeremenko.ua@ieee.org.

Oleksandr Lemeshko, Dr.Sc., Professor, Professor of the Department of Infocommunication Engineering, Kharkiv National University of Radio Electronics. Research interests: Traffic Management, Optimization in Telecommunications, Routing, Traffic Engineering, QoS, QoE, Hierarchical Routing, Fault-Tolerant Routing. Address: Ukraine, 61166, Kharkiv, Nauka Ave., 14, Phone/fax: +380577021320, e-mail: oleksandr.lemeshko.ua@ieee.org.

Oleksandra Yeremenko, Cand.Sc., Associate Professor, Senior Researcher, Doctoral Candidate of the Department of Infocommunication Engineering, Kharkiv National University of Radio Electronics. Research interests: NGN, Future Internet, Quality of Service, Traffic Management, Fault-Tolerant Routing, Network Security. Address: Ukraine, 61166, Kharkiv, Nauka Ave., 14, Phone/fax: +380577021320, e-mail: oleksandra.yeremenko.ua@ieee.org.

СИСТЕМЫ И ПРОЦЕССЫ УПРАВЛЕНИЯ

УДК 519.713

МЕТОД ЗНИЖЕННЯ ЕКОЛОГІЧНОЇ НЕБЕЗПЕКИ ПРОМИСЛОВОГО ОБ'ЄКТА ЧУБ І.А., НОВОЖИЛОВА М.В.

Розглядається задача зниження рівня екологічної небезпеки системи, що містить джерела забруднення атмосфери аерозольними викидами пожежі. Описуються етапи оптимізаційного методу розв'язання задачі розміщення пожежонебезпечних об'єктів з мінімізацією величини концентрації забруднюючих домішок у множині точок контролю.

Ключові слова: оптимальне розміщення, екологічна безпека, пожежонебезпечний об'єкт.

Key words: optimal placement, environmental safety, fire-hazard object.

1. Вступ

Одним з важливих показників ступеня ефективності управління промисловим господарством регіону як складною організаційно-технічною системою є оцінка рівня техногенної безпеки території [1,2], що обов'язково включає екологічну складову. Передумовою забезпечення прийнятного за європейськими стандартами рівня екологічної безпеки навколишнього середовища є раціональне розміщення джерел забруднення з урахуванням кліматичних факторів та суміжних рекреаційних зон. Раціональне планування забудови території проводиться на етапі проектування, і цей процес вимагає побудови оптимізаційної прогнозової моделі та визначення оптимального варіанту розміщення екологічно небезпечних об'єктів. Аналіз множини надзвичайних ситуацій техногенного характеру, що сталися за останні роки на території України, показав, що значний внесок в екологічну небезпеку території вносять пожежі, які внаслідок своєї масовості та великої кількості викидів забруднюючих аерозольних та газоподібних продуктів горіння в атмосферу порушують екологічний баланс. Тому об'єктом даного дослідження є специфічний клас екологічно небезпечних об'єктів, а саме пожежонебезпечні об'єкти, пожежі на яких є джерелами забруднюючих аерозольних викидів в атмосферу. В [3-5] побудовано та досліджено оптимізаційну математичну модель розміщення пожежонебезпечних об'єктів з мінімізацією величини концентрації забруднюючих домішок у критичних зонах. Проведено моделювання можливої зони забруднення, показано, що задача оптимального розміщення пожежонебезпечних об'єктів за певних умов може бути зведена до оптимізаційної задачі розміщення їх зон забруднення.

Метою даної роботи є побудова оптимізаційного аналітичного методу розв'язання задачі розміщення пожежонебезпечних об'єктів з мінімізацією величини концентрації забруднюючих домішок у множині точок контролю.

2. Виклад основного матеріалу

Розглянемо оптимізаційну задачу розміщення множини S пожежонебезпечних об'єктів $S=\{S_i\}$, $i=\overline{1,N}$ в обмеженій в загальному випадку багатозв'язній області розміщення Ω . При цьому кожен об'єкт S_i у випадку надзвичайної ситуації техногенного характеру, що супроводжується пожежею, породжує в області Ω зону T_i забруднення атмосфери аерозольними домішками, яка апроксимується восьмикутником, орієнтованим по розі вітрів.

Тоді постановка оптимізаційної задачі є такою: визначити розміщення пожежонебезпечних об'єктів S_i , $i=\overline{1,N}$, в області Ω , щоб максимальна сумарна концентрація аерозольних викидів можливих пожеж у скінченній множині точок контролю $P=\{p_\zeta\}=\{x_\zeta^p, y_\zeta^p\}$, $\zeta=\overline{1,D}$, була мінімальною, тобто знайти

$$\min_{(x_i, y_i) \in D} \max_{(x_\zeta^p, y_\zeta^p) \in P} \sum_{i=1}^N C(x_\zeta^p, y_\zeta^p, x_i, y_i, G_i, Q), \quad (1)$$

де $C(\dots)$ – функція концентрації забруднюючих домішок, що досягається викидами з N джерел; $G=(g_1, g_2, \dots, g_k)$ – вектор геометричних та теплофізичних параметрів пожежі; $Q=(q_1, q_2, \dots, q_n)$ – вектор параметрів, що задають природно-кліматичні умови області Ω .

Результатом розв'язання задачі (1) є вектор

$$u=(x_1, y_1, x_2, y_2, \dots, x_N, y_N) \in D \subset R^{2N}$$

параметрів розміщення (координат полюсів) пожежонебезпечних об'єктів.

Область допустимих розв'язків D формується системою обмежень, яка включає геометричні обмеження на параметри розміщення об'єктів T_i і фізичні обмеження на характеристики результуючого поля $C(\dots)$ приземних концентрацій аерозольних викидів пожежі в області Ω .

Задача (1) є NP-повною оптимізаційною задачею умовного дискретно-неперервного програмування, тому отримання точного розв'язку за прийнятний час у загальному випадку гарантувати не можна. Отже, пропонується метод пошуку раціональних розв'язків і їх перебір, в результаті якого визначається наближене значення локального екстремуму функції мети.

Метод розв'язання оптимізаційної задачі (1) складається з таких основних етапів:

- визначення початкового варіанту розміщення пожежонебезпечних об'єктів S в області Ω ;
- пошук вектора u^* , що відповідає локальному

мінімуму функції мети

$$F(u) = \max_{(x_i^p, y_i^p) \in P} \sum_{i=1}^N C(x_i^p, y_i^p, x_i, y_i, G_i, Q); \quad (2)$$

• перебір локальних мінімумів функції мети $F(u)$. При цьому виділяється рекордне значення функції мети і визначається відповідний йому вектор u^{**} параметрів розміщення об'єктів.

Розглянемо етапи методу детальніше.

Задача визначення початкового варіанту розміщення пожежонебезпечних об'єктів в області Ω формулюється так:

знайти допустимий вектор $u^0 = (x_1^0, y_1^0, \dots, x_N^0, y_N^0) \in D$ параметрів

розміщення об'єктів T_i , $i = \overline{1, N}$, в області Ω .

Для розв'язання даної задачі використано евристичний підхід, заснований на методі послідовно-одиначного розміщення плоских орієнтованих геометричних об'єктів із змінними метричними характеристиками [6].

Таким чином, метод пошуку початкового розміщення складається з N ітерацій, і на кожній ітерації розв'язується задача виду:

$$\text{знайти: } u_i^0 \in D_{0_i} \subset R^2,$$

де підобласть допустимих рішень $D_{0_i} \subset D$ формується обмеженнями тільки на параметри розміщення об'єкта S_i , всі інші об'єкти вважаються нерухомими.

Відзначимо, що задача визначення початкового варіанту розміщення пожежонебезпечних об'єктів в області Ω не є оптимізаційною.

Ітераційний процес пошуку початкового розміщення об'єктів $\{T_i\}$ в області Ω містить таку послідовність кроків:

1. Визначення геометричних характеристик восьмикутників T_i , якими моделюються зони забруднення аерозольними викидами можливих пожеж на об'єктах S_i , $i = \overline{1, N}$.

2. Визначення послідовності розміщення джерел викидів. Джерела S_i , $i = \overline{1, N}$, (пожежонебезпечні об'єкти) ранжуються за величиною площі зони забруднення T_i , заданої для умов плоскої горизонтальної рівнини і параметрів вітру, що відповідає розі вітрів незбуреного метеорологічного вітру. Першим розміщується джерело, що має найбільшу площу зони забруднення.

3. Побудова допустимої області розміщення об'єкта T_i , $i = 1, 2, \dots, N$. Оскільки в області Ω присутні зони заборони, то область на кожному кроці рішення є багатозв'язною. Система обмежень включає такі умови:

$$\Phi_{i0}(x_i, y_i, m_0) \geq 0, \quad (3)$$

$$\Phi_{ij}(x_i, y_i, x_j, y_j) \geq 0, \quad (4)$$

$$\Phi_{ic}(x_i, y_i, x_c, y_c) \geq 0, \quad (5)$$

$$C(x, y, u, G, Q)|_{(x,y) \in P} + C_\Phi \leq \Gamma_{ДК}, \quad (6)$$

$$L_{i\mu} \geq (\leq) L_{i\mu}^*, \quad i = \overline{1, N}, \quad i \neq j, \quad \mu = \overline{1, M}. \quad (7)$$

де $\Phi_{i0}(x_i, y_i, m_0)$ – Φ -функція [6] об'єкта T_i і області $\overline{\Omega} = cl(R^2 \setminus \Omega)$; m_0 – метричні характеристики Ω ; $\Phi_{ij}(x_i, y_i, x_j, y_j)$ – Φ -функція об'єктів T_i і T_j , $i, j = \overline{1, N}$, C_Φ – фонові концентрації, $L_{i\mu}$; $(L_{i\mu}^*)$ – технологічні обмеження на розміщення S_i щодо елементів Σ_μ транспортних мереж, інженерних комунікацій та максимальні (мінімальні) допустимі відстані між S_i та Σ_μ відповідно.

При $i = 1$ (розміщення першого об'єкта) в області Ω відсутні інші об'єкти розміщення, тому система обмежень буде включати в себе лише умови (3), (5) – (7).

4 Визначення параметрів розміщення об'єкта T_i в області Ω .

5. Повторення ітерацій 2 - 3 для об'єктів, що є нерозміщеними.

Перейдемо до другого етапу методу розв'язання задачі (1), а саме до визначення локального мінімуму функції мети (1). Вектор u^0 параметрів розміщення об'єктів T_i є початковою точкою алгоритму розв'язання оптимізаційної задачі (1) з обмеженнями на область D припустимих значень вигляду (3) – (7).

Для розв'язання задачі використовується схема методу мінімізації за групами змінних, що представляють собою координати полюса об'єкта T_i , $i = \overline{1, N}$, який має розміщуватися на поточній ітерації.

Введемо позначення:

r – номер ітерації, кожна ітерація складається з N кроків, де N – число розміщуваних об'єктів;

$u_i^p = (x_i^p, y_i^p)$ – параметри розміщення об'єкта T_i на r -й ітерації;

$u^p = (u_1^p, u_2^p, \dots, u_N^p)$ – вектор координат полюсів об'єктів на r -й ітерації.

Загальна схема алгоритму наближення до локального екстремуму є такою.

Крок 1. За деяким правилом визначається об'єкт T_i , що має на r -й ітерації параметри розміщення

$$u_i^p = (x_i^p, y_i^p).$$

Крок 2. Виділяються обмеження, що формують в околі полюса об'єкта T_i область D_i^p допустимих параметрів його розміщення на r -й ітерації.

Крок 3. Визначаються нові параметри розміщення об'єкта T_k , такі, що

$$(u_1^p, u_2^p, \dots, u_k^{p+1}, \dots, u_N^p) \in D_k^p \quad i$$

$$F(u_1^p, u_2^p, \dots, u_i^{p+1}, \dots, u_N^p) \leq F(u_1^p, u_2^p, \dots, u_i^p, \dots, u_N^p).$$

Крок 4. Якщо $u_i^{p+1} = u_i^p$, то $i = i + 1$. При $i \leq N$ здійснюється повернення до кроку 1.

Крок 5. Якщо $u^{p+1} \neq u^p$, то переходять до $(p + 1)$ -ї ітерації. В іншому випадку параметри розміщення об'єктів, отримані на p -ї ітерації, вважаються розв'язком задачі.

Зауваження 1. Зсув об'єкта T_i (крок 3) виконується в напрямку антиградієнта функції $F(u)$ за параметрами розміщення T_i . В цьому випадку конкретні алгоритми, що реалізують наведену загальну схему, можуть відрізнятися правилом визначення рухомого об'єкта T_i (крок 1) і вибором величини кроку його зсуву.

Розглянемо докладніше роботу алгоритму на p -ї ітерації. Вона починається з обчислення модулів градієнта цільової функції $F(u)$ (2) за параметрами u :

$$M_i^p = |\text{grad} F(u)| = \left[\left(\frac{\partial F(u)}{\partial x_i} \right)^2 + \left(\frac{\partial F(u)}{\partial y_i} \right)^2 \right]^{1/2}.$$

З причини відсутності аналітичного виразу для функції $F(u)$ її частинні похідні по параметрах розміщення об'єктів визначаються чисельно.

Зауваження 2. Результируюча концентрація забруднюючої домішки від N джерел в області Ω визначається напрямком і швидкістю приземного вітру в точках розміщення джерел.

Визначення множини $M^p = (M_1^p, M_2^p, \dots, M_N^p)$ на p -ї ітерації супроводжується впорядкуванням компонент по зростанню. Відповідно до цього порядку відбувається зміна розміщення об'єктів з набору об'єктів T .

Припустимо, найбільше значення з множини M^p має компонента M_1^p , якій відповідає об'єкт T_i . В цьому випадку першим змінюється місце розміщення об'єкта T_i . У напрямку антиградієнта цільової функції $F(u)$ за параметрами $u_i^p = (x_i^p, y_i^p)$ розміщення об'єкта T_i робиться крок, який не виводить за межі D_1^p області допустимих параметрів його розміщення на p -ї ітерації.

Для цього визначається величина $\alpha_i^p \geq 0$ така, що

$$(z_1^p, z_2^p, \dots, z_i^p - \alpha_i^p g_{ix}^p, \dots, z_N^p) \in D_1^p,$$

де $g_k^p = \text{grad} F(u)$ за параметрами розміщення об'єкта T_i на p -ї ітерації.

Координати нового положення полюса (параметри розміщення) об'єкта T_i знаходяться із співвідношень

$$u_i^{p+1} = u_i^p - \alpha_i^p g_i^p,$$

або в проекціях g_{ix}^p і g_{iy}^p вектора градієнта g_i^p на координатні вісі:

$$x_i^{p+1} = x_i^p - \alpha_i^p g_{ix}^p, \quad y_i^{p+1} = y_i^p - \alpha_i^p g_{iy}^p.$$

Величина кроку α_i^p може визначатися різними способами, однак для зручності комп'ютерної

реалізації запропонованого алгоритму величину вибираємо постійною.

Зазначений алгоритм дозволяє отримати раціональне розміщення об'єктів. Це пояснюється тим, що об'єкти розміщуються послідовно. Однак вибір такого методу розв'язання, що є модифікацією методу Гаусса-Зейделя, обумовлений особливостями даної задачі і дозволяє отримати досить хороше наближення до локального мінімуму за прийнятний час.

При цьому мінімізація в просторі розмірності $2N$ замінюється N -кратною мінімізацією в двовимірному просторі параметрів розміщення об'єктів T_i , $i=1, N$. Це дозволило скоротити кількість громіздких обчислень, пов'язаних з необхідністю перевірки виконання системи обмежень і зменшити число обмежень на кожному кроці оптимізації.

Значення локального мінімуму функції мети задачі (1) залежить від вектора початкового розміщення об'єктів u^0 . Тому в процесі пошуку розв'язку необхідно здійснити перебір початкових точок і відповідних їм локальних мінімумів, в процесі якого знаходиться екстремальне значення функції мети і вектор u^{**} координат полюсів пожежонебезпечних об'єктів S_i (і їх зон забруднення T_i), $i=1, N$, який відповідає мінімуму максимальної концентрації викидів пожеж в точках контролю. Кількість локальних мінімумів, що перебираються, може визначатися різними шляхами, наприклад, особою, яка приймає рішення, виходячи з умов і обмежень конкретної задачі, виконання певних вимог тощо.

3. Висновки

Розглянуто етапи та проведено дослідження методу розв'язання оптимізаційної задачі розміщення пожежонебезпечних об'єктів у заданій області з урахуванням впливу небезпечних чинників можливої пожежі на навколишнє середовище та кліматичних умов, а також моделювання можливої форми зони забруднення земної поверхні викидами пожежі.

Запропонований математичний апарат програмно реалізовано та застосовано для розв'язання практичних задач реконструкції цеху з виготовлення спиртових настоїв ТОВ «Панацея», м. Запоріжжя, при проектуванні АЗС на території ЗАТ «АвтоАЗ», м. Запоріжжя, а також при реконструкції Гнідинцівського газопереробного заводу, м. Варва Чернігівської області, про що є відповідні акти впровадження.

Література: 1. Попов В.М., Чуб І.А., Новожилова М.В. Показатели эффективности региональной системы техногенной безопасности // Вестник Командно-инженерного института МЧС Республики Беларусь. 2014. № 2(20). С. 32-41. 2. Попов В.М., Чуб І.А., Новожилова М.В. Модель адаптивной системы техно-

генной безопасности региона // Системы управління, навігації та зв'язку. 2013. Вип. 2(26). С. 120-123.

3. *Новожилова М.В., Чуб І.А.* Условия размещения экологически опасного объекта с учетом изменения геометрических характеристик области возможного загрязнения // Науковий вісник будівництва. 2009. № 54. С. 240-244.

4. *Чуб І.А., Новожилова М.В., Андронов В.А.* Моделирование размещения пожароопасных объектов: минимизация загрязнения атмосферы. Харків: ІД ІНЖЕК. 2012. 133 с.

5. *Чуб І.А.* Оптимізація розміщення екологічно небезпечних об'єктів // Радиоелектроника и информатика. 2017. №3. С. 12-14.

6. *Стоян Ю. Г., Яковлев С.В.* Математические модели и оптимизационные методы геометрического проектирования. К.: Наук. думка, 1986. 267 с.

Transliterated Bibliography:

1. *Popov V.M., Chub I.A., Novozhylova M.V.* Pokazately efektyvnosti regyonal'noj systemu tehnogennoj bezopasnosti // Vestnyk Komandno-ynzhenernogo ynstytuta MChS Respublyky Belarus'. 2014. № 2(20). С. 32-41.
2. *Popov V.M., Chub Y.A., Novozhylova M. V.* Model' adaptivnoj systemu tehnogennoj bezopasnosti regyona // Systemy upravlinnja, navigacii' ta zv'jazku. 2013. Vyp. 2(26). С. 120-123.
3. *Novozhylova M.V., Chub I.A.* Uslovyja razmeshhenja ekologicheskyy opasnogo ob'ekta s uchetom yzmenenija geometrycheskyyh harakterystyk oblasti vozmozhnogo zagryaznenija // Naukovyj visnyk budivnytstva. 2009. № 54. С. 240-244.
4. *Chub Y.A., Novozhylova M.V., Andronov V.A.* Modelyrovanye razmeshheniya pozharoopasnykh ob'ektov: mynymyzacyja zagryaznenija atmosferu. Harkiv: ID INZhEK. 2012. 133 s.
5. *Chub I.A.* Optymizacija rozmishhennja ekologichno nebezpechnyy ob'ektiv // Radioelectronics & Informatics. 2017. #3. С. 12-14.
6. *Stojan Ju. G., Jakovlev S.V.* Matematycheskyye modely y optymyzacyonnue metody geometrycheskogo proektyrovanyja. K.: Nauk. dumka, 1986. 267 s.

Надійшла до редколегії 02.02.2017

Рецензент: д-р фіз.-мат. наук, проф. Сізова Н.Д.

Чуб Ігор Андрійович, д-р техн. наук, професор, начальник кафедри пожежної профілактики в населених пунктах Національного університету цивільного захисту України, м. Харків. Наукові інтереси: математичне моделювання та оптимізація структури та функціонування складних організаційно-технічних систем. Адреса: Україна, 61000, Харків. вул. Чернишевська, 94, тел. (057) 707-34-13.

Новожилова Марина Володимирівна, д-р фіз.-мат. наук, професор, завідувач кафедри економічної кібернетики та інформаційних технологій Харківського національного університету будівництва та архітектури. Наукові інтереси: економіко-математичне моделювання та теорія та методи дослідження операцій. Адреса: Україна, 61000, Харків, вул. Сумська, 40, тел. (057) 706-20-49.

Chub Igor, Doctor of Sci., Professor, Head of department of fire prevention in the settlements of the National University of Civil Defence of Ukraine, Kharkov c. Research interests are mathematical modeling and optimization of the structure and functioning complex technical systems. Address: Kharkov c., st. Chernyshevskaya, 94, tel. (057) 707-34-13.

Novozhylova Maryna, Doctor of Sci., Professor, Head of the department of economic cybernetics and information technologies of Kharkov National University of Civil Engineering and Architecture. Research interests are economic modeling and the theory and methods of operations research. Address: Kharkov c., Sumskaya str., 40, tel. (057) 706-20-49.

КОМПЬЮТЕРНАЯ ИНЖЕНЕРИЯ

УДК 658:512.011; 681.326; 519.713

КВАНТОВЫЕ МОДЕЛИ И ОБЛАЧНЫЕ СЕРВИСЫ ДЛЯ АНАЛИЗА И ДИАГНОСТИРОВАНИЯ ЛОГИЧЕСКИХ СХЕМ

ЕМЕЛЬЯНОВ И.В., ЛЮБАРСКИЙ М.М.,
ХАХАНОВ В.И.

Предлагаются логические структуры киберфизического компьютеринга, которые рассматриваются как компоненты облачных технологий для точного мониторинга и метрического управления объектами. Дается аналитический обзор киберфизических технологий, задекларированных в Gartner's Hype Cycle 2017, а также некоторые разъяснения, связанные с их применением в науке, образовании, транспорте, промышленности и государственных структурах. Описывается кубитный метод диагностирования неисправностей и минимизации логических схем, которые отличаются от аналогов параллельным выполнением логических операций. Предлагаются кубитные структуры данных, облачная реализация методов анализа, тестирования и моделирования цифровых систем. Даются рекомендации к использованию топ 10 компонентов суперцикла 2017 в бизнесе и научно-образовательном процессе университетов. Представляется memory-driven архитектура квантового компьютеринга, которая характеризуется использованием фотонных транзакций на структуре электронов при отсутствии логики, связанной с суперпозицией и перепутыванием состояний.

Ключевые слова: hype cycle emerging technologies, digital world, cyber-physical computing, machine learning, cloud-driven computing, quantum computing, qubit data structures.

1. Введение

Целью исследования является обоснование и разработка архитектуры квантового компьютеринга на основе использования кубитных структур данных для параллельного решения задач синтеза и анализа цифровых устройств.

Задачи: 1) Анализ современных киберфизических трендов развития цифровых технологий управления физическими и социальными процессами, представленных компанией Gartner. 2) Модернизация архитектуры квантового memory-driven компьютеринга без операций суперпозиции и перепутывания. 3) Метод трансформирования табличных моделей цифровых компонентов в кубитные структуры данных. 4) Методы квантовой минимизации, облачного анализа и диагно-

стирования булевых функций путем унитарного кодирования входных состояний.

Киберфизический компьютеринг представляет собой теорию и практику точного облачного управления виртуальными, социальными и физическими процессами и явлениями на основе использования больших данных, метрического онлайн-мониторинга цифрового состояния процесса или явления в целях улучшения качества жизни человека и сохранения экологии планеты. Существуют определенные тенденции в мире, которые создают технологическую основу для создания киберсоциального компьютеринга, как части киберфизического компьютеринга, в рамках технологического уклада Internet of Things. Компания Gartner Inc., предсказывающая технологическую кибермоду, в 2017 году добавила восемь новых трендов в свой бренд Hype Emerging Technologies Cycle: 5G, Artificial General Intelligence, Deep Learning, Deep Reinforcement Learning, Digital Twin, Edge Computing, Serverless PaaS and Cognitive Computing [1,2].

Edge computing является технологией повышения производительности облачных сервисов путем выполнения локальных вычислений по месту мобильного пользователя. Digital Twin создает киберобразы физических процессов и явлений. Как в зеркале, если нет отображения оцифрованной компании (университета) в киберпространстве, то ее нет в физическом пространстве. Serverless PaaS – бессерверная архитектура для организации вычислительных процессов на основе платформы как услуги (Platform as a Service). Экономика облачной платформы является явно выигрышной по сравнению с серверной поддержкой деятельности компаний. Поэтому весь малый и средний бизнес в течение двух лет перейдет на облачные инфраструктуры и сервисы.

Как следует понимать фазы Gartner цикла? 1) Innovation Trigger – запуск инновации, где потенциально интересные для рынка прорывные технологии с еще недоказанной коммерческой состоятельностью идут на смену существующим киберфизическим укладам. 2) Peak of Inflated Expectations – пик раздутых рыночных ожиданий, где своевременная реклама создает успешные прецеденты создания инновационных технологий на фоне множества неудач. 3) Trough of Disillusionment – приход разочарования, когда интерес к технологиям угасает, эксперименты не подтверждают ожидаемую рыночную привлека-

тельность, отдельные разработчики улучшают свою продукцию и получают инвестиции. 4) *Slope of Enlightenment* – склон прозрения, когда появляются примеры технологий, приносящие пользу предприятию, находятся финансы для пилотных проектов. 5) *Plateau of Productivity*: плоскость устойчивого повышения производительности, когда создаваемые технологии, товары и услуги находят своего потребителя на международном рынке.

2. Три главных направления киберкультуры

Hype-cycle 2017 формирует киберкультуру планеты на последующие 5-10+ лет путем экспертного анализа более, чем 1800 возможных технологий, выполняемого ведущими исследовательскими и консалтинговыми компаниями.

Первые три места в Gartner-топ-цикле закреплены за следующими стратегическими направлениями: *Artificial Intelligence Everywhere*, *Transparently Immersive Experiences* и *Digital Platforms*.

1) *Artificial Intelligence Everywhere*. Искусственный интеллект становится самой дизрапторной технологией в последующие 10 лет благодаря наличию вычислительных мощностей, бесконечных объемов больших данных и достижений в реализации нейронных сетей для адаптации к новым ситуациям, с которыми никто и никогда не сталкивался ранее. Предприятия, которые заинтересованы в использовании искусственного интеллекта, считают полезными для себя следующие технологии: Deep Learning, Deep Reinforcement Learning, Artificial General Intelligence, Autonomous Vehicles, Cognitive Computing, Commercial UAVs (Drones), Conversational User Interfaces, Enterprise Taxonomy and Ontology Management, Machine Learning, Smart Dust, Smart Robots and Smart Workspace.

Таким образом, Artificial General Intelligence в последующие 10 лет будет проникать во все сферы человеческой деятельности, как технологическая услуга, погруженная в киберфизическое пространство, включая 30 процентов высокотехнологичных и транспортных компаний.

Smart Workspace – умное рабочее место означает быть подключенным к инфраструктуре решения производственных проблем в пространстве и времени по формату 24/7. При этом используются виртуальные частные сети, метрика измерения потенциала и результатов деятельности, наличие определенной киберкультуры и выбор наиболее удобных мест для ведения бизнеса. Высокая самомотивация к успешному и результативному выполнению задания обуславливает использование динамически изменяющегося киберфизического рабочего пространства для творчества, инвариантного к офису, дому, транспорту, местам отдыха и спорта.

2) *Transparently Immersive Experiences*. Технологии опыта прозрачного погружения становятся все более ориентированными на человека и обеспечивают: 1) прозрачность отношений между людьми, бизнесом и вещами; 2) гибкость и адаптивность связей между рабочим местом, домом, предприятием и другими людьми. Gartner-Inc. также предсказывает внедрение в практику следующих, ожидаемых всеми, критических технологий: Autonomous Vehicles, Brain-Computer Interfaces, Smart Dust, 4D Printing, Augmented Reality (AR), Connected Home, Human Augmentation, Nanotube Electronics, Virtual Reality (VR), and Volumetric Displays. Интеграция кибер-технологий направлена на обеспечение качества жизни человека путем создания: smart workspace, connected home, augmented reality, virtual reality and the growing brain-computer interface. Напри-мер, Human Augmentation технология направлена на расширение или дополнение человеческих возможностей в целях улучшения здоровья и качества жизни за счет гармоничного использования когнитивных и биотехнических улучшений, как частей человеческого тела. Volumetric Displays, как объемные дисплеи, визуализируют объекты с помощью 3D активных элементов-вокселей (voxels) в трех измерениях со сферическим углом обзора в 360 градусов, где изображение явления изменяется при перемещении зрителя. Технология 4D Printing является инновацией 3D-печати, где конструктивные материалы могут трансформироваться после производства изделия в целях адаптации продукта к потребностям человека и к окружающей среде.

3) *Digital Platforms*. Ключевые платформы технологической культуры формируются компонентами: 5G, Digital Twin, Edge Computing, Blockchain, IoT, Neuromorphic Hardware, Quantum Computing, Serverless PaaS и Software-Defined Security. Такие технологии, как Quantum Computing и Blockchain, будут создавать самые непредсказуемые и дизрапторные прорывы для человека в ближайшие 5-10 лет. Neuromorphic Hardware рассматривается как будущее искусственного интеллекта, которое направлено на создание нейроморфного компьютерного чипа, способного заменить облачные вычислительные мощности Apple Siri Data Center при решении сложных задач machine learning (Chris Eliasmith, a theoretical neuroscientist and co-CEO of Canadian AI startup Applied Brain Research) [3]. Иначе, внутри iPhone появится цифровой мозг в форме нейроморфного IP-core, оперативно и на месте решающий все задачи взаимодействия гаджета с внешним миром в реальном времени. Нейроморфный универсальный чип IBM, благодаря

спайковому асинхронизму, потребляет на три порядка меньше энергии при количестве транзисторов, превышающих в пять раз существующие аппаратные решения компании Intel. Для программирования аппаратно-ориентированных алгоритмов используются компиляторы: Nengo, Python. Путем использования компилятора Nengo сегодня уже реализованы цифровые системы на кристаллах: vision systems, speech systems, motion control, adaptive robotic controllers, а также Spaun-chip для автономного интерактивного общения компьютера с окружающей средой. Software-Defined Security (SDS) или Catbird предназначена для защиты системных объектов или логических структур в виртуальном пространстве. Это связано с тем, что сетевая безопасность уже не имеет физических границ в рамках существования логической архитектуры облачных сервисов. Поэтому создается точная и гибкая SDS в виде дополнения к инфраструктурам и центрам обработки данных без наличия специализированных аппаратных устройств защиты. Масштабирование SDS дает возможность создавать или приобретать минимально необходимые условия безопасности в определенном месте и времени, что существенно уменьшает материальные затраты на формирование качественного SDS сервиса.

3. Практика использования топ-технологий

Высокие затраты на исследования и разработки от Amazon, Apple, Baidu, Google, IBM, Microsoft и Facebook стимулируют создание оригинальных патентуемых решений в области Deep Learning и Machine Learning, среди которых следует отметить: Amazon Alexa, Apple Siri, Google Now, Microsoft Cortana. Компания Gartner Inc. уверена, что инструменты для глубокого обучения будут составлять 80% стандартных средств для ученых к 2018 году. Сегодня уже на сайтах компаний становятся доступными технологии и данные о научных исследованиях: Amazon Machine Learning, Apple Machine Learning Journal, Baidu Research, Google Research, IBM AI и Cognitive Computing, Facebook Research.

Внедрение 5G-технологии телекоммуникаций в ближайшее десятилетие предоставит рынку ожидаемые инновационные решения по безопасности, масштабируемости и производительности глобальных сетей и соединений в транспорте, IoT, индустрии, здравоохранении.

Gartner Inc. прогнозирует, что к 2020 году 3% сетевых провайдеров услуг мобильной связи запустят коммерческие сети в 5G-формате, что обеспечит качественно новые условия повсеместного внедрения телекоммуникаций для масштабируемой глобализации сервисов: IoT, cloud-transport control, UHD-телевидения. лиде-

рами 5G-внедрения в 2017-2018 году выступают: AT&T, NTT Docomo, Sprint USA, Telstra, T-Mobile и Verizon. Технология 5G представляет собой ультраширокополосную мобильную связь в миллиметровом диапазоне для Massive M2M транзакций в реальном времени с допустимыми для управления задержками (1мс), при одновременном подключении порядка 10 млн устройств на 1 кв км. 5G использует технологию множественного доступа с разделением луча (Beam Division Multiple Access – BDMA) для взаимодействия базовой станции с мобильными устройствами. Беспроводная сотовая архитектура 5G обеспечивает пропускную способность 10-50 Гбит/с в миллиметровом диапазоне частот 30-300 ГГц для приложений UHD видео и создания виртуальной реальности [4]. Инновационная технология 5G характеризуется использованием: массива приемо-передающих антенн Massive MIMO, сети Cognitive Radio, организацией непосредственной связи D2D для IoT, созданием сети радиодоступа, как облачной услуги (radio access network as a service) и облака виртуальных сетевых функций (network function virtualization cloud – NFV).

Green IoT – киберфизическая культура человеческой деятельности, направленная на обеспечение качества жизни людей и сохранение экологии планеты, энергии, ресурсов и времени. Компонентами IoT являются: Identification, Sensing, Controlling, Communication, Computation, Services Intelligent, Digital Infrastructure. Умный мир (smart world) предоставляет каждому человеку сервисы от: умных устройств (watches, mobile phones, computers), умного транспорта (aircrafts, cars, buses, trains), умной инфраструктуры (homes, hospitals, offices, factories, cities, states), умного образования (school, university).

Компания Gartner Inc. посчитала необходимым вывести из-под зонтика актуальной рыночной моды следующие технологии, как не оправдавшие ожидания IT-бизнеса: Affective Computing, Micro Data Centers, Natural-Language Question Answering, Personal Analytics, Smart Data Discovery and Virtual Personal Assistants.

Для создания успешных бизнесов и новых образовательных курсов компания Gartner Inc. рекомендует учитывать свои предположения о стратегическом планировании, которые включают 10 пунктов: 1) К 2020 году 100 миллионов потребителей будут делать покупки в расширенной реальности, в том числе с использованием Head-Mounted Displays (HMDs). 2) К 2020 году 30% сеансов просмотра веб-страниц будут выполняться без использования экрана. Более 5 из 550 миллионов обладателей Apple iPhone будут использовать AirPods для обмена голосовыми со-

общениями. Пять процентов веб-сайтов, ориентированных на потребителя, будут оснащены аудиоинтерфейсами (включая голосовые чаты с поддержкой голоса). 3) К 2019 году 20% брендов откажутся от своих мобильных приложений (в пользу MASA – Mesh App and Service Architecture). 4) К 2020 году умные алгоритмы положительно повлияют на поведение более 1 миллиарда глобальных работников. 5) К 2022 году бизнес на основе использования блокчейнов будет стоить 10 миллиардов долларов. 6) К 2021 году 20% всех видов деятельности человека будут включены, по крайней мере, в сервисы одной из семи ведущих глобальных компаний (Google, Apple, Facebook, Amazon, Baidu, Alibaba и Tencent). 7) До 2019 года каждый доллар, инвестированный в инновации, потребует дополнительных 7 долларов для основного исполнения проекта. 8) В течение 2020 года Internet of Things (IoT) на 3% увеличит спрос, связанный с data centers. Комнатные экранные устройства, такие как Amazon Echo и Google Home, будут находиться в более, чем 10 миллионах домов. 9) К 2022 году IoT и облака (Google, Amazon, Microsoft) будут экономить потребителям и предприятиям 1 триллион долларов в год, ориентированных на услуги и расходные материалы. К 2020 году около 40 миллионов автомобилей будут использовать Android Auto, а 37 миллионов транспортных средств будут использовать CarPlay. 10) К 2020 году 40% сотрудников смогут сократить свои расходы на здравоохранение, используя фитнес-трекер.

4. Инновации для архитектуры квантового компьютеринга

Физическая основа классического квантового компьютеринга (рис. 1) заключается в использовании операций суперпозиции и перепутывания над состояниями электронов (p), которых вполне достаточно для организации вычислительного процесса [11-13]. Электрон выполняет функцию памяти для хранения бита информации. Транзакции между электронами осуществляются с помощью квантов или фотонов (q). Низкая и высокая орбиты электрона соответствуют значениям нуля и единицы.

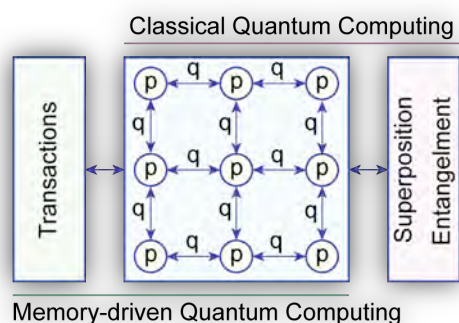


Рис. 1. Два вида квантового компьютеринга

Функционально полный базис для создания архитектуры квантового компьютеринга представлен операциями суперпозиции и перепутывания, которым можно поставить в соответствие традиционный логический базис или-не. В теории множеств данному базису ставится в соответствие изоморфизм в форме пары «объединение-дополнение». Однозначно, операция суперпозиции в квантовой физике изоморфна логической инверсии или теоретико-множественному дополнению в алгебре логики. Поэтому естественно, что каждое состояние двоичного разряда (электрона) после применения данной операции к биту информации знает друг о друге все, где бы они не находились, с точностью до инверсии $a_i / \bar{a}_i$. Далее на основе упомянутой пары операций-примитивов (или, не) строится более сложная система логических компонентов и устройств для организации и оптимизации вычислительных процессов. Недостатки квантового классического компьютеринга: 1) Высокая стоимость поддержания температурных условий для функционирования квантовых атомарных структур на уровне -270 градусов Цельсия. 2) Наблюдаемость результатов вычислительных процессов, приводящих к разрушению данных после их чтения.

Инновация в архитектуре квантового компьютеринга определяется устранением логики, связанной с суперпозицией и перепутыванием. Аналогом могут выступать memory-driven архитектуры классического компьютера, свободные от reusable logic. В таком компьютере нет ничего, кроме памяти, где реализуется транзакция (операции записи-считывания данных) на адресуемой памяти. Транзакции достаточно для организации любого вычислительного процесса путем использования единственного характеристического уравнения [11, 13]: $M_i = Q_i[M(X_i)]$.

Здесь M_i , Q_i , X_i представляют собой компоненты памяти: для вектора-состояния вычислительного процесса; вектора-кубита логического примитива; вектора-адреса ячейки логического примитива. Q-логика реализуется на адресной памяти, где также осуществляется ассемблирование всех примитивов с помощью интегрирующего M-вектора состояния вычислительного устройства, который формирует двоичные адреса $M(X)$ на основе использования массива номеров входных переменных X .

Инновационное предложение заключается в создании квантового memory-driven компьютеринга без квантовых операций суперпозиции и перепутывания (или, не) на основе использования приведенного выше характеристического уравнения, задающего две транзакции записи-считывания на

структуре электронов (см. рис. 1). Исключить две сложные операции из квантового компьютеринга – означает существенно упростить архитектуру и привести ее к структуре памяти на электронах для выполнения транзакций между ними с помощью квантов или фотонов.

Подтверждением состоятельности предложенной инновационной квантовой архитектуры может служить несколько свежих публикаций, которые фиксируют устойчивую тенденцию к созданию квантового компьютеринга на атомарной структуре памяти с передачей информации с помощью фотонов или квантов.

Ученые из Калифорнийского технологического института создали оптическую квантовую память [14], в которой информация передается путем кодирования данных с использованием квантового состояния фотонов. Память реализована на редкоземельных элементах и способна сохранять состояния фотонов с помощью резонаторов-посредников между атомом и светом. Размерность квантовой памяти в 1000 раз меньше, чем традиционные классические решения. Она реализована в нанополости, которая позволяет хранить информацию в очень небольшом объеме.

Практическая реализация идеи замены электронов фотонами приводит к созданию компьютеринга с быстродействием, близким к скорости света [15]. Корейские исследователи сделали еще один шаг к оптическим вычислениям. Они создали photon-triggered нанопроволочный транзистор на основе кристаллического и пористого кремния, где переключение и усиление величины токового сигнала осуществляется под воздействием фотона. Использование фотонов в логических вентилях AND, OR и NAND приведет к ультракомпактным нанопроцессорам и наноразмерным фотоприемникам для получения изображений с высоким разрешением.

Ученые из Колумбийского университета провели успешные опыты по созданию транзистора из одного атома в молекулярной электронике [16]. Они реализовали геометрически упорядоченный кластер неорганических атомов с центральным ядром, состоящим из 14 атомов, которое связали с золотыми электродами, что позволило управлять транзистором под воздействием одного электрона при комнатной температуре.

Впервые достигнута передача цифровых сигналов между молекулами, что является существенным достижением на пути развития молекулярного компьютеринга [17]. Создание электронных компонентов из отдельных молекул является многообещающей стратегией для миниатюризации и интеграции электронных устройств. Однако практическая реализация молекулярных устройств и схем для передачи и обработки сиг-

налов при комнатной температуре оказалась сложной задачей, которая решена путем размещения молекул SnCl_2Pc на поверхности меди (Cu). Плоскостная ориентация молекул в межмолекулярном взаимодействии служит носителем информации. В связанных молекулярных массивах сигнал передается от одной молекулы к другой по наперед заданным маршрутам, реализующим логические операции. Явления плоскостной ориентации позволяют использовать молекулы, имеющие внутренние бистабильные состояния, для создания сложных молекулярных устройств и схем.

Теоретическое сходство классического компьютеринга с квантовым заключается в общей модели вычислительной архитектуры, которая использует: память для хранения данных и функционально полный базис примитивных элементов (или, не) = (суперпозиция, перепутывание) для реализации арифметико-логических операций над данными.

Каковы же формальные различия между классическим и квантовым компьютерингом? Первый из них последовательно обрабатывает адресуемые или упорядоченные гетерогенные данные, затрачивая на процедуру $Q=n$ тактов. Он также способен обработать гомогенные данные параллельно и за один автоматный такт. Если данные не упорядочены и представляют собой множества, то предельная вычислительная сложность их обработки на классическом компьютере зависит от мощности двух множеств и определяется как $Q = n \times m$. Например, для пересечения двух множеств: $M_1 \cap M_2 = \{Q, E, H\} \cap \{E, H, J\} = \{E, H\}$ необходимо затратить 6 автоматных тактов. Квантовый компьютеринг устраняет данный недостаток, связанный с квадратичной или мультипликативной вычислительной сложностью процедуры пересечения на классическом компьютере. Он решает задачу одновременной и параллельной обработки теоретико-множественных данных. Примером тому может служить параллельное выполнение приведенной выше операции пересечения над множествами за один автоматный такт. Для этого предварительно выполняется операция суперпозиции или объединения примитивных символов, входящих в множества $M_1 \cap M_2 = \{V\} \cap \{C\} = \{P\} = \{E, H\}$, но при этом используется замкнутый теоретико-множественный алфавит [10]: $B^*(Y) = \{Q, E, H, J, O = \{Q, H\}, I = \{E, J\}, A = \{Q, E\}, B = \{H, J\}, S = \{Q, J\}, P = \{E, H\}, C = \{E, H, J\}, F = \{Q, H, J\}, L = \{Q, E, J\}, V = \{Q, E, H\}, Y = \{Q, E, H, J\}, U = \emptyset\}$. Символы алфавита представляют собой множество всех подмножеств на универсуме Y , которые составлены путем суперпозиции примитивов.

Квантовая суперпозиция дает возможность сосредоточить в одной точке гильбертова пространства несколько дискретных состояний. Аналогично операция объединения также создает в одной точке дискретного пространства символический образ, содержащий несколько состояний. Исходя из сказанного, достаточно просто использовать многозначный замкнутый алфавит для моделирования квантовых вычислений на классическом компьютере. Но для этого необходимо предварительно создать символическую систему или алгебру множеств для кодирования состояний. Самой простой является алгебра Кантора, которая оперирует двумя дискретными состояниями и создает 4 символа: $A^k = \{0, 1, X = \{0, 1\}, \emptyset\}$. Символика данного алфавита с точностью до изоморфизма представляет собой теоретико-множественную интерпретацию кубита. Иначе, суперпозиция двух состояний одного кубита образует 4 символа. Естественно, что два кубита способны сгенерировать 16 состояний, три кубита – 64 состояния. В общем случае число состояний Q имеет зависимость от числа кубитов n , которая представлена следующей формулой: $Q = 2^{2^n}$.

Для параллельного выполнения, но уже логических операций над кубитами интерес представляет кодирование примитивных символов алфавита унарным двоичным кодом. Остальные символы получаются суперпозицией кодов примитивов. Исключение составляет код символа пустого множества, который получается путем применения операции логического умножения (пересечения). Для алгебры Кантора таблица соответствия “Symbol – Code” имеет следующий вид:

$a_i \in A^k$	0	1	X	$\emptyset$
$C(a_i)$	10	01	11	00

Платой за параллелизм выполнения логических операций над множествами в классическом компьютере является существенное увеличение разрядности (регистра, памяти) для унарного кодирования примитивных символов алфавита. Аналогичная таблица соответствия для кодирования 16-ричного алфавита $B^*(Y)$ на четырех унарных кодах примитивов имеет вид:

$a_i \in B^*$	Q	E	N	J	O	I	A	B	S	P	C	F	L	V	Y	$\emptyset$
$C(a_i)$	1	0	0	0	1	0	1	0	1	0	0	1	1	1	1	0
	0	1	0	0	0	1	1	0	0	1	1	0	1	1	1	0
	0	0	1	0	1	0	0	1	0	1	1	1	0	1	1	0
	0	0	0	1	0	1	0	1	1	0	1	1	1	0	1	0

Что касается табличной модели логического элемента, он изначально представлен квантоподобной совокупностью равнозначных строк или множеством дискретных отношений между

входными и выходными переменными. Взамен такого множества предлагается кубитный вектор выходных состояний, ориентированный на адресное параллельное моделирование цифровых логических схем. Замена неупорядоченного множества равнозначных строк таблицы истинности на упорядоченный вектор адресуемых состояний дает возможность создавать параллельный компьютеринг на классических вычислителях за счет увеличения объема памяти для унарного кодирования каждого состояния. Иначе, суперпозиция n равнозначных элементов конечного множества в квантовом (Q) компьютеринге имеет взаимно-однозначное соответствие n -мерному вектору в классическом (C) адресуемом компьютеринге (рис. 2). Данный вектор получен путем выполнения og-операции над унарными кодами примитивных элементов исходного множества. Естественно, что любое пересечение (and), объединение (or) или дополнение (not) в C-компьютере унарных кодов данных выполняется параллельно, за один автоматный такт, как и в Q-компьютере. Платой за полученное быстродействие является увеличение объема памяти (количества битов) для унарного кодирования символов относительно позиционного кодирования, которое определяется следующим выражением: $Q = n^2 / n \times \log n = n / \log n$.

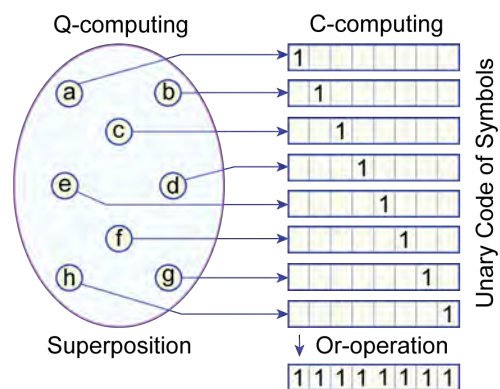


Рис. 2. Суперпозиция элементов и логическое объединение векторов

Таким образом, расширению мощности теоретико-множественного алфавита можно поставить в соответствие наращивание кубитов в квантовом компьютере. Это дает возможность параллельно и в одном автоматном такте осуществлять вычислительные процессы на основе использования логических (теоретико-множественных) операций. Суперпозиции n элементов конечного множества в квантовом (Q) компьютеринге взаимно-однозначно соответствует n -мерный вектор в классическом (C) адресуемом компьютеринге, который получается на основе использования og-операции над унарными кодами символов ис-

ходного множества, выполняемой параллельно за один автоматный такт.

Интерес представляет оптимальное решение проблемы покрытия путем использования цифровой регистровой структуры (Quantum Coverage Processor – QCP), создающей все возможные сочетания входных векторов в форме кубитных структур данных, представленных булеаном или множеством всех подмножеств (рис. 3).

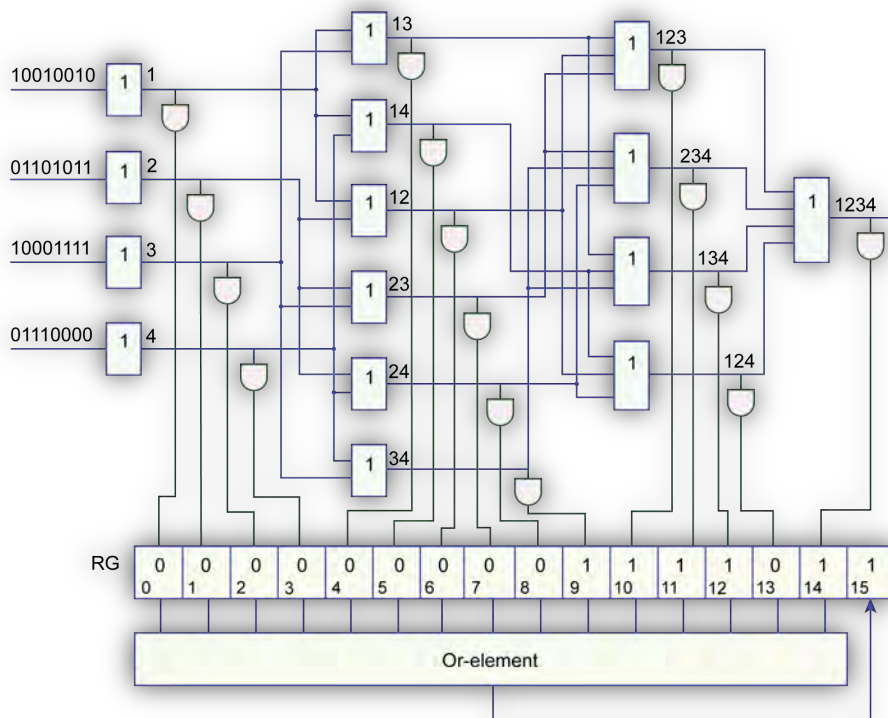


Рис. 3. QC-Processor для определения оптимального покрытия

Схема имеет логический анализатор при каждом регистровом выходе, который определяет полноту покрытия путем выполнения and-функции по всем разрядам регистровой переменной. Число таких функций-преобразователей “вектор-бит” соответствует количеству элементов и равно $Q = 2^n - 1$. Битовые результаты вычисления and-функций интегрируются в регистр анализа RG, который своими единичными разрядами идентифицирует получение покрытия. Последний разряд регистра, равный 1, свидетельствует о существовании позитивного результата, полученного в процессе поиска покрытия. При этом минимальное покрытие определяется крайней левой единицей в регистре-анализаторе RG. Схема также предназначена для определения примитивизма или уникальности входных векторов, что идентифицируется нулевыми значениями по всем разрядам регистра-анализатора, кроме последнего, равного в этом случае единице. Аппаратная сложность регистровой цифровой схемы для поиска оптимального покрытия, где n – ко-

личество строк, m – длина регистра, определяется следующим аналитическим выражением:

$Q = [2^{n+1} + 2^n] \times m + 2^n$. Вычислительная сложность совокупных процедур при поиске оптимального покрытия равна, в худшем случае, n автоматным тактам. Вместо og-операций в логических элементах можно использовать хог-функции, что позволит решать задачи по иденти-

фикации и распознаванию киберобъектов, представленных в векторной форме. Здесь используется аксиома: хог-операция двух векторов-примитивов создает их логическое объединение или суперпозицию. Таким образом, n объектов в дискретном векторном пространстве распознаются, если все хог-сочетания, кроме последнего, формируют нулевые значения на выходе and-анализаторов.

Пример. Определить минимальное покрытие единичными значениями восьми разрядов следующей таблицы:

X – Inputs	0	1	2	3	4	5	6	7
1	1	0	0	1	0	0	1	0
2	0	1	1	0	1	0	1	1
3	1	0	0	0	1	1	1	1
4	0	1	1	1	0	0	0	0

Воздействие на схему четырьмя регистровыми переменными: $X = (1, 2, 3, 4)$ формирует следующее состояние регистра анализатора:

RG = 0000000001111011

Единичные значения регистра показывают существование четырех возможных вариантов решения задачи: $C = \{3, 4\}$, $\{1, 2, 3\}$, $\{2, 3, 4\}$, $\{1, 3, 4\}$, $\{1, 2, 3, 4\}$. Минимальное покрытие обеспечивается двумя входными векторами: $C = \{3, 4\}$, которое идентифицируется левой крайней единицей в регистре-анализаторе RG.

QCP-структура может быть упрощена путем устранения регистра, который выполняет функцию хранения результатов поиска оптимального покрытия. В этом случае QCP-схема становится строго логической, где позитивный результат поиска определяется уже за один автоматный такт единичным значением состояния выхода интегрального og-элемента. Оптимальное покрытие будет идентифицироваться единичным зна-

чением выхода and-элемента, который топологически ближе находится к внешним входам схемы.

5. Метод квантовой минимизации булевых функций

Квантовое представление данных в виде суперпозиции унарных кодов можно использовать для существенного упрощения метода неопределенных коэффициентов при минимизации булевых функций.

Утверждение. Любая сколь угодно сложная таблица истинности дискретного объекта может быть представлена не более, чем двумя векторами квантового покрытия при унитарном кодировании (UC) входных состояний. Процедура, иллюстрирующая данное утверждение, показана на рис. 4. Здесь представлены нулевые и единичные кубы таблицы истинности – входные состояния, которые унарно кодируются и логически объединяются (VUC). В результате получаются два вектора квантового покрытия, где каждый из которых может представлять логическую функцию в форме кубитного покрытия.

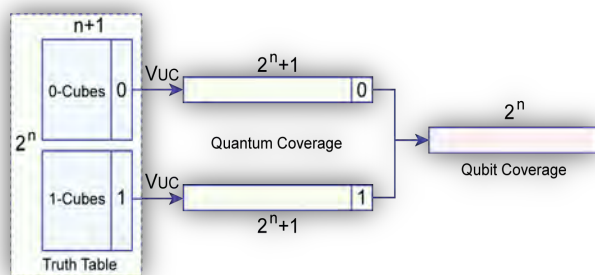


Рис. 4. Модель получения кубитного покрытия

Пусть имеется таблица $T = T_{ij}$, $i = \overline{1, m}$; $j = \overline{1, n}$. В худшем случае число состояний в каждом столбце таблицы (матрицы) равно m . Данное количество состояний может быть представлено m рядами унитарного кода. Например, два символа алфавита Кантора можно представить одной строкой (11), состоящей из двух двоичных разрядов:

$$A^k = \{0 \rightarrow 10; 1 \rightarrow 01; X \rightarrow 11; \emptyset \rightarrow 00\}.$$

В результате получается матрица размерностью m на n , где каждая ячейка содержит m -разрядный вектор. Применение операции суперпозиции или логического объединения ко всем строкам таблицы, благодаря унитарному кодированию, создает одну строку, которая формирует в компактном виде отношения, ранее представленные исходной таблицей. Например, все двоично-десятичные коды входных состояний логического элемента от трех переменных представляются вектором (11111111), если учитывать следующее кодирование: 000 – 10000000, 001 – 01000000, 010 – 00100000, 011 – 00010000, 100 – 00001000,

101 – 00000100, 110 – 00000010, 111 – 00000001. Однако таблица истинности представляет собой функциональное соответствие:

$$Y = f(X), X \rightarrow Y, X = \{x_1, x_2, \dots, x_i, \dots, x_n\}, Y = \{0, 1\}.$$

Учитывая, что функция определена на двух дискретных значениях $\{0, 1\}$, то двоичная по выходам таблица истинности может быть всегда представлена двумя строками, каждая из которых суперпозиционирует или собирает нулевые или единичные унитарные коды входных воздействий. Например, таблица истинности хог-элемента на три входа после логического объединения унарных кодов входных состояний по единичному и нулевому значению выхода будет иметь следующий вид:

a	b	c	Y	a	b	c	U-code	Y
0	0	0	0	0	0	0	10000000	0
0	0	1	1	0	0	1	01000000	1
0	1	0	1	0	1	0	00100000	1
0	1	1	0	0	1	1	00010000	0
1	0	0	1	1	0	0	00001000	1
1	0	1	0	1	0	1	00000100	0
1	1	0	0	1	1	0	00000010	0
1	1	1	1	1	1	1	00000001	1

$$\begin{matrix} V \\ \forall Y=1 \\ V \\ \forall Y=0 \end{matrix}$$

0	1	1	0	1	0	0	1	1
1	0	0	1	0	1	1	0	0

Таким образом, любая таблица истинности цифрового устройства может быть представлена в явном виде двумя строками-кубами квантового покрытия, которые суперпозиционно объединяют единичные и нулевые (по состоянию выхода) входные воздействия исходной таблицы. Учитывая, что кубы квантового покрытия всегда взаимно инверсны, то для задания функциональности достаточно оставить один из них, полагая, что второй можно быстро и просто достроить с помощью операции инверсии, в случае необходимости.

С учетом использования полученной информации о возможности представления таблицы истинности двумя кубами квантового покрытия далее предлагается модернизация известного метода неопределенных коэффициентов для минимизации логических функций. Пусть имеется исходная таблица неопределенных коэффициентов для минимизации булевой функции от трех переменных [21]:

T_{ij}	$x_1 x_2 x_3$	k_1	k_2	k_3	k_{12}	k_{13}	k_{23}	k_{123}	f_i
0	000	k_1^0	k_2^0	k_3^0	k_{12}^{00}	k_{13}^{00}	k_{23}^{00}	k_{123}^{000}	1
1	001	k_1^0	k_2^0	k_3^1	k_{12}^{00}	k_{13}^{01}	k_{23}^{01}	k_{123}^{001}	0
2	010	k_1^0	k_2^1	k_3^0	k_{12}^{01}	k_{13}^{00}	k_{23}^{10}	k_{123}^{010}	1
3	011	k_1^0	k_2^1	k_3^1	k_{12}^{01}	k_{13}^{01}	k_{23}^{11}	k_{123}^{011}	0
4	100	k_1^1	k_2^0	k_3^0	k_{12}^{10}	k_{13}^{10}	k_{23}^{00}	k_{123}^{100}	1
5	101	k_1^1	k_2^0	k_3^1	k_{12}^{10}	k_{13}^{11}	k_{23}^{01}	k_{123}^{101}	0
6	110	k_1^1	k_2^1	k_3^0	k_{12}^{11}	k_{13}^{10}	k_{23}^{10}	k_{123}^{110}	0
7	111	k_1^1	k_2^1	k_3^1	k_{12}^{11}	k_{13}^{11}	k_{23}^{11}	k_{123}^{111}	1

Данная таблица преобразуется к двоичному виду всех возможных комбинаторных сочетаний состояний входных переменных, которые способны сформировать значения выходов функции, представленной в последнем столбце:

T _{ij}	x ₁ x ₂ x ₃	1	2	3	12	13	23	123	f
0	000	0	0	0	00	00	00	000	1
1	001	0	0	1	00	01	01	001	0
2	010	0	1	0	01	00	10	010	1
3	011	0	1	1	01	01	11	011	0
4	100	1	0	0	10	10	00	100	1
5	101	1	0	1	10	11	01	101	0
6	110	1	1	0	11	10	10	110	0
7	111	1	1	1	11	11	11	111	1

Естественно, что полученные в ячейках таблицы комбинации входных воздействий: 0,1; 00, 01, 10, 11; 000, 001, 010, 011, 100, 101, 110, 111 три-виально трансформируются в унитарные коды двоичных состояний 10, 01; 1000, 0100, 0010, 0001; 10000000, 01000000, 00100000, 00010000, 00001000, 00000100, 00000010, 00000001 соответственно:

T _{ij}	x ₁ x ₂ x ₃	1	2	3	12	13	23	123	f
0	000	10	10	10	1000	1000	1000	10000000	1
1	001	10	10	01	1000	0100	0100	01000000	0
2	010	10	01	10	0100	1000	0010	00100000	1
3	011	10	01	01	0100	0100	0001	00010000	0
4	100	01	10	10	0010	0010	1000	00001000	1
5	101	01	10	01	0010	0001	0100	00000100	0
6	110	01	01	10	0001	0010	0010	00000010	0
7	111	01	01	01	0001	0001	0001	00000001	1

Далее выполняется раздельное логическое объединение всех единичных и нулевых строк таблицы в два интегрирующих вектора. В результате получают все возможные сочетания переменных, которые формируют единичные и нулевые значения функции:

Q	Operations	1	2	3	12	13	23	123	f
1	$Q^1 = \bigvee_{f_i=1} T_{ij}$	11	11	11	1111	1011	1011	10101001	1
2	$Q^0 = \bigvee_{f_i=0} T_{ij}$	11	11	11	1111	0111	0111	01010110	0
3	$Q = (\bigvee_{f_i=1} T_{ij}) \wedge (\bigvee_{f_i=0} T_{ij})$	00	00	00	0000	1000	1000	10101001	Y

Чтобы получить дизъюнктивную форму минимизированной размерности (строка 3 в приведенной выше Q-таблице), необходимо из единичного куба квантового покрытия логически вычесть нулевой куб по правилу, представленному в следующей формуле:

$$Q = (\bigcup_{f_i=1} T_{ij}) \setminus (\bigcup_{f_i=0} T_{ij}) = (\bigvee_{f_i=1} T_{ij}) \wedge (\overline{\bigvee_{f_i=0} T_{ij}}).$$

Дешифрирование полученного квантового куба Q в дизъюнктивную нормальную форму дает следующий результат:

$$Y = \bar{x}_1 \bar{x}_3 \vee \bar{x}_2 \bar{x}_3 \vee \bar{x}_1 \bar{x}_2 \bar{x}_3 \vee x_1 x_2 \bar{x}_3 \vee x_1 \bar{x}_2 \bar{x}_3 \vee x_1 x_2 x_3.$$

Данная форма не является минимальной, поэтому нуждается в применении средств, позволяющих решить задачу покрытия простейших исходных единичных термов (000, 010, 100, 111) полученными решениями. Для функции Y очевидно, что первые два терма (0x0, x00) покрывают логические слагаемые 3,4,5 или (000, 010, 100), которые, в данном случае, являются избыточными в соответствии с правилом поглощения $a \vee ab = a$, что дает возможность записать минимальную дизъюнктивную нормальную форму в следующем виде:

$$Y = \bar{x}_1 \bar{x}_3 \vee \bar{x}_2 \bar{x}_3 \vee x_1 x_2 x_3.$$

Другое решение задачи покрытия связано с применением QC-процессора, имеющего 6 регистровых входов, что позволит определить минимальную ДНФ путем моделирования двоичных кодов-строк $X_i \in X$ таблицы покрытия:

T	000	010	100	111	$X_i \in X$
0x0	1	1	.	.	1100
x00	1	.	1	.	1010
000	1	.	.	.	1000
010	.	1	.	.	0100
100	.	.	1	.	0010
111	.	.	.	1	0001

Результат моделирования кодов определяет в качестве минимального покрытия три кода-строки, которые создают минимальную функцию:

$$Y = 1100 \vee 1010 \vee 0001 \rightarrow 0x0 \vee x00 \vee 111 \rightarrow \bar{x}_1 \bar{x}_3 \vee \bar{x}_2 \bar{x}_3 \vee x_1 x_2 x_3.$$

Данный метод целесообразно использовать для получения минимальной ДНФ или КНФ по таблицам истинности, где количество нулевых и единичных кубов-строк не сильно отличается друг от друга. Другое применение метода связано с существенной минимизацией дефектной области при поиске неисправностей в цифровых системах.

Вычислительная сложность Q квантового метода неопределенных коэффициентов определяется выражением, которое формирует время для унарного кодирования состояний таблицы истинности (для сравнения Q^b – есть сложность базового метода минимизации):

$$Q = 2^n \times 3^n;$$

$$Q^b = 2^n \times 2^n \times 2^n = 2^n \times 2^{n+n};$$

$$R = \frac{Q}{Q^b} = \frac{2^n \times 3^n}{2^n \times 2^{n+n}} = \frac{3^n}{2^{2n}}.$$

Таким образом, вычислительная сложность Q получения компактного квантового покрытия для минимизации булевых функций существенно

меньше по сравнению с базовым методом Q^b неопределенных коэффициентов, использующим специальной формы таблицу истинности. Исключив предварительную обработку таблицы истинности, которая заключается в унарном кодировании состояний, вычислительная сложность собственно метода минимизации булевых функций включает всего три векторных параллельных операции.

Затраты памяти H для хранения структур данных формируются размерностью таблицы, необходимой для суперпозиционного получения двух векторов квантового покрытия, где ячейки таблицы представлены унарными кодами состояний:

$$\begin{aligned} H &= 2^n \times 3^n; \\ H^b &= 2^n \times 2^n = 2^{2n}; \\ S &= \frac{H}{H^b} = \frac{2^n \times 3^n}{2^n \times 2^n} = \frac{3^n}{2^n}. \end{aligned}$$

Таким образом, чтобы получить компактное квантовое покрытие, необходимо использовать таблицу H существенно большей размерности по сравнению с исходной таблицей истинности H^b . Предложен квантовый способ минимизации булевых функций, который отличается от метода неопределенных коэффициентов параллельным выполнением операции суперпозиции над нулевыми и единичными состояниями входных переменных, представленными унитарными кодами, что дает возможность существенно повысить быстродействие за счет избыточной памяти.

6. Кубитный метод поиска дефектов

Рассматривается кубитный метод поиска дефектов путем теоретико-множественной разности двух векторов, соответствующих единичному и нулевому значению состояний выходов, как реакций наблюдаемых выходов на входной тест проверки неисправностей:

$$F = \left(\bigcup_{\forall R_i=1} Q_{ij} \right) \setminus \left(\bigcup_{\forall R_i=0} Q_{ij} \right) = \left(\bigvee_{\exists R_i=1} Q_{ij} \right) \wedge \left(\overline{\bigvee_{\exists R_i=0} Q_{ij}} \right).$$

Структуры данных представлены таблицей неисправностей на декартовом произведении тестовых наборов и множестве линий объекта диагностирования, где каждая ячейка представляет собой два бита: первый из них идентифицирует проверяемую константную неисправность нуля (10), а второй – константную неисправность единицы (01):

$$\begin{aligned} Q &= \{F, T, L\}, \\ Q &= Q_{ij}, i = 1, m; j = 1, n; \\ F &= (F_1, F_2, \dots, F_j, \dots, F_n), \\ F_j &= \{10 = 0; 01 = 1; 11 = \{= 0, = 1\}; 00 = \emptyset\}; \\ T &= (T_1, T_2, \dots, T_i, \dots, T_m); \\ L &= (L_1, L_2, \dots, L_j, \dots, L_n). \end{aligned}$$

Суперпозиция неисправностей (две единицы на одной линии-ячейке) дает возможность существенно минимизировать структуры данных для хранения информации в целях последующего поиска дефектов при выполнении диагностического эксперимента в режиме online.

Для проверки метода поиска дефектов далее предлагается логическая схема, представленная на рис. 5, которая имеет 6 элементов and-not, 11 линий, 5 входов и 2 выхода.

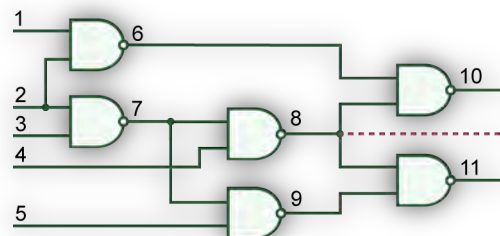


Рис. 5. ISCAS-схема для верификации

Следующая таблица иллюстрирует выполнение диагностического эксперимента в целях объединения множества дефектов, которые формируют некорректные состояния выходов на тестовых наборах $\{T1-R10; T5-R11; T6-R10, R11; T8-R11\}$:

$Q = Q_{ij}$	1	2	3	4	5	6	7	8	9	10	11	R_{10}	R_{11}
T_1	01	10	01	00	10	00	10	10	00	10	01	1	0
T_2	10	00	10	00	01	10	00	00	10	01	10	0	0
T_3	00	01	01	00	00	01	10	01	01	10	10	0	0
T_4	10	00	01	00	10	00	01	00	10	01	01	0	0
T_5	00	10	00	01	00	01	00	10	00	10	10	0	1
T_6	01	10	00	00	10	00	00	01	10	01	10	1	1
T_7	01	00	00	10	00	00	01	00	10	01	01	0	0
T_8	00	10	10	01	01	10	00	00	00	01	10	0	1
Q_1	01	11	11	01	11	11	10	11	10	11	11	1	1
Q_0	11	01	11	10	11	11	11	01	11	11	11	0	0
F	00	10	00	01	00	00	00	10	00	00	00	1/0	1/0

Здесь дизъюнкция строк T_1, T_5, T_6, T_8 формирует вектор Q_1 , который собирает все возможные дефекты, проверяемые на тестовых наборах. Вектор Q_0 с помощью строк T_2, T_3, T_4, T_7 объединяет все невозможные, непроверяемые на тестовых наборах дефекты. Вычитание всех невозможных из всех возможных дефектов дает искомый результат в виде трех неисправностей, закодированных как $F_2 = 10; F_4 = 01; F_8 = 10$. Таким образом, параллельное выполнение двух регистровых or -операций на основе результатов проведенного диагностического эксперимента позволило определить три возможных неисправности, каждая из которых может иметь место в логической схеме:

$$F = \{2^0, 4^1, 8^0\}.$$

Более жестким является ограничительное условие существования в логической схеме одиноч-

ного константного дефекта, использование которого приводит к вычислению дефектов на основе следующего выражения:

$$F = \left(\bigcap_{\forall R_i=1} Q_{ij} \right) \setminus \left(\bigcup_{\forall R_i=0} Q_{ij} \right) = \left(\bigwedge_{\exists R_i=1} Q_{ij} \right) \wedge \left(\bigvee_{\exists R_i=0} \overline{Q_{ij}} \right).$$

Применение данной формулы существенно уточняет результат диагностирования и приводит его к виду: $F = \{2^0\}$ из-за противоречивости кодов дефектов по and-операции в столбцах 4 и 8. Условие наличия в логической схеме одиночной константной неисправности ставит во главу угла следующее утверждение.

Утверждение. Если в столбце таблицы неисправностей существует координата 00 или 01, которая создает на наблюдаемых выходах некорректность $R=1$, связанную с неисправностью 10 на остальных координатах столбца, то такой одиночный дефект (10) в логической схеме невозможен.

Д о к а з а т е л ь с т в о. Пусть на p тестовых наборах зафиксировано несовпадение на внешних выходах эталонных и реальных значений сигналов. При этом $p-1$ координата в рассматриваемом столбце имеет значение 10 (01) и лишь одна p -координата имеет значение 01 (10). Если предположить, что в логической схеме имеется дефект 10, то на p -координате также должен присутствовать дефект 10, который создает некорректное состояние выходов. Но по условиям моделирования такой дефект там отсутствует. Следовательно, невозможно считать, что в схеме присутствует дефект 10. Это подтверждается также формальным результатом – пустым пересечением всех координат столбца, связанных с некорректными состояниями выходов схемы:

$$F = \left(\bigwedge_{\exists R_i=1} Q_{ij} \right) = \begin{cases} 10 \wedge 10 \wedge \dots \wedge 10 \wedge 01 = 00; \\ 10 \wedge 10 \wedge \dots \wedge 10 \wedge 00 = 00. \end{cases}$$

Все сказанное относится и к состоянию p -координаты, которое идентифицируется сигналом пустого множества 00, взаимодействие с которым также делает невозможным присутствие в логической схеме одиночной константной 0-неисправности (код 10).

7. Cloud Service «Quantum Modeling»

Предлагается облачный сервис проектирования и тестирования цифровых устройств, который отличается от существующих аналогов эргономичным интерфейсом быстрого ввода структурно-функционального графического описания в виде рисунка, а также адресно-ориентированным кубитно-векторным моделированием исправного поведения и неисправностей в целях существенного повышения быстродействия интерпретативного анализа, синтеза тестов и диагностирования.

Цель – предоставление облачных сервисов проектирования и тестирования цифровых устройств широкому кругу специалистов и студентов, связанному с внедрением и отработкой новой технологии квантового моделирования на основе использования кубитных структур данных.

Задачи: 1) Структуры кубитных данных для проектирования модели цифрового устройства. 2) Структура функциональных компонентов облачного сервиса «Quantum Modeling». 3) Тестирование сервиса моделирования на реальных цифровых логических схемах. 4) Облачная имплементация сервисов моделирования и тестирования логических схем. 5) Практические результаты и выводы.

8. Структуры кубитных данных цифрового устройства

Можно ли отказаться от структурно-логической схемы описания модели при синтезе и анализе цифрового изделия? Ответ связан с методо- driven реализацией логических или комбинационных схем. Для этого необходимо создавать новые оригинальные форматы визуализации, нетрадиционные по отношению к исторически сложившимся шаблонам проектирования компьютерных компонентов. Далее рассматривается пример логической схемы (рис. 6) из библиотеки международного симпозиума ISCAS, для которой выполняется построение структур данных.

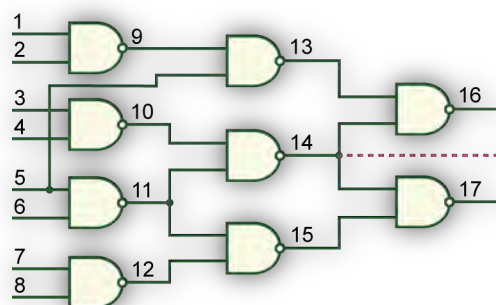


Рис. 6. Структура логических элементов из библиотеки ISCAS

Одним из возможных вариантов инновационной формы описания цифровых логических схем является суперпозиционная структура кубитных векторов, представленная на рис. 7. Здесь результат анализа определяется суперпозицией кубитных векторов, создающих вектор моделирования M , где алгоритм анализа содержит одну процедуру – сдвига векторов вверх-вниз относительно вектора моделирования. Виртуальный сдвиг технологически просто реализуется путем вычисления адресов ячеек кубитных векторов $M_i = Q_i[M(X_i)]$, из которых считывается информация и заносится в вектор моделирования исправного поведения. Структурная взаимосвязь

кубитных векторов-примитивов осуществляется посредством нумерации логических переменных: входных, внутренних и выходных. Переменные формируют своими двоичными состояниями адреса ячеек кубитных векторов для вычисления реакции цифрового устройства на входное воздействие (01001110 110101110).

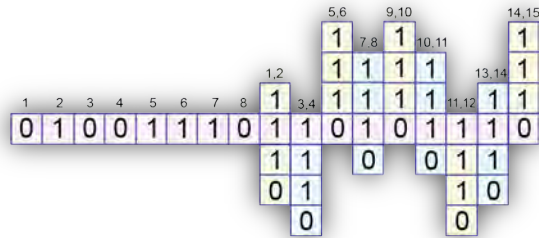


Рис. 7. Инновационные кубитные структуры данных логической схемы

Структуры кубитных данных для синтеза и анализа цифровых систем и компонентов оперируют следующей моделью описания логических схем:

$$\begin{aligned} S &= \{M, X, Y, Q\}, \\ M &= (M_1, M_2, \dots, M_i, \dots, M_m), \\ X &= (X_1, X_2, \dots, X_i, \dots, X_n), \\ Y &= (Y_1, Y_2, \dots, Y_i, \dots, Y_k), \\ Q &= (Q_1, Q_2, \dots, Q_i, \dots, Q_q), \\ M_i &= Q_i[M(X_i)]. \end{aligned}$$

Здесь представлены следующие системные компоненты: M – вектор моделирования цифрового устройства, который связывает все кубитные покрытия примитивных элементов в структуру; X – вектор входных переменных, заданных двоичными значениями; Y – вектор выходных переменных, которые формируют реакцию цифрового устройства; Q – кубитное покрытие, представленное в виде вектора выходных состояний логического элемента и предназначенное для формирования его функции. Основное характеристическое уравнение для моделирования цифрового устройства оперирует вычислением адресов для записи-считывания данных, что создает простые и быстродействующие транзакции между вектором моделирования и кубитными покрытиями:

$$M_i = Q_i[M(X_i)].$$

Чтобы определить двоичное значение логической переменной или линии M_i , необходимо сформировать адрес ячейки кубитного покрытия, который создается конкатенацией двоичных состояний вектора моделирования M , где адреса ячеек вектора задаются номерами-идентификаторами входных переменных X . Характеристическое уравнение непосредственно влияет на быстродействие квантового метода моделирования, которое зависит от операций

конкатенации k , считывания $г$ и записи w битов, количества q кубитных покрытий в цифровой схеме или логических примитивов, а также длины теста (входных наборов) t :

$$Q = (k + r + w) \times q \times t.$$

Таким образом, всего три транзакционных операции необходимы для обработки логического элемента любой функциональной сложности!

9. Структура облачного сервиса «Quantum Modeling»

Современная стратегия компьютерного бизнеса заключается в массовом переводе всех процессов и явлений из чисто физического в киберфизическое пространство. Это делает сервисы менее уязвимыми для киберпреступности, экономически выгодными и технологически доступными в пространстве и во времени, без каких-либо ограничений. Поэтому высокотехнологический бизнес Design and Test, определяемый компаниями, формирующими индекс капитализации NASDAQ, усиленными темпами уходит в облачный киберфизический компьютеринг, который имеет следующую примерную структуру, представленную на рис. 8.

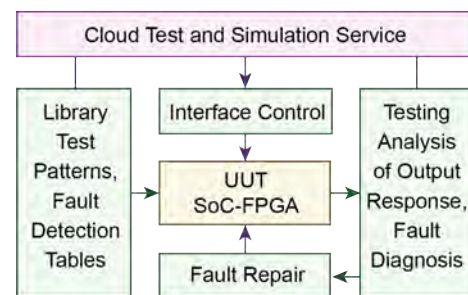


Рис. 8. Design and Test Cloud Online Service

Здесь имеется только один физический модуль с IP-адресом, Unit Under Test (любые цифровые устройства, сенсоры, актюаторы, мобильные устройства, автомобильные компьютеры, конечные гаджеты), который находится в физическом пространстве. Остальные блоки, указанные в структуре, создают облачные сервисы, доступные в режиме онлайн 24/7. Таким образом, любое физическое устройство может быть легко продиагностировано, благодаря его подключению к облачным сервисам в режиме автономного тестирования или даже в режиме рабочего функционирования. Все сказанное относится и к образовательному процессу, когда студенты могут воспользоваться сколь угодно сложной аппаратурой, подключенной через интернет к собственному гаджету для получения киберфизических сервисов, связанных с выполнением реальных технологических экспериментов на компьютерных устройствах.

Облачный сервис «Quantum Modeling» реализован на языке SWIFT (2300 строк кода) и содержит следующие структурные компоненты или модули: 1) Q-function – кубитная модель функционального примитива, входящая в состав логической схемы. 2) Data Structures – структура цифрового устройства, основанная на использовании сквозной нумерации входных внутренних и выходных переменных, которые фигурируют в качестве вход-выходных идентификаторов линий логических элементов. 3) Interface – интерфейс связи цифрового устройства с внешним окружением создает контакты для воздействия на изделие внешними функциональными или тестовыми наборами, а также для снятия двоичной информации с внешних наблюдаемых выходов схемы. 4) Run-Step – функциональные сервисы моделирования входных воздействий в режимах: пошаговое моделирование теста путем ручного задания или ввода двоичного входного набора; автоматическое моделирование исчерпывающего теста на всех 2^n входных последовательностях. 5) Visual – сервис визуализации схемы цифрового устройства в составе: логических элементов; соединительных линий между элементами, внешними входами и выходами; визуализация внешних входных и выходных контактов; визуализация состояний логических элементов на одном входном наборе и на полном тесте; визуализация вектора моделирования на одном тестовом наборе и на полном тесте; визуализация нумерации логических элементов, входных и выходных портов, а также десятичных кодов, соответствующих двоичным кубитным векторам логических примитивов. 6) Infrastructure – инфраструктурные сервисы для обслуживания основных функций, такие как: сохранение файла со схемным описанием цифрового устройства; коррекция схемного описания путем удаления-рисования структурных компонентов, кубитных покрытий и межэлементных связей. 7) Optimizer – оптимизация визуального размещения схемных компонентов и связей путем применения волнового алгоритма и ограничений, связанных с минимальным расстоянием между линиями и элементами. 8) Fault Simulation – моделирование неисправностей цифрового устройства на основе использования кубитных покрытий логических элементов и булевых производных, дающих возможность одновременной обработки списков неисправностей по дедуктивному алгоритму. 9) Test Synthesis – синтез тестовых наборов для проверки одиночных константных неисправностей путем взятия булевых производных по кубитному покрытию для всех входных переменных цифрового устройства на основе встречного сдвига и логического суммирования симметрич-

ных соседних частей кубитного вектора. Screen-Shot экрана, который визуализирует интерфейс сервиса «Quantum Modeling», представлен на рис. 9.

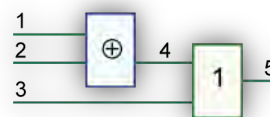


Рис. 9. Визуализация экрана с сервисом Quantum Modeling

Следующая схема (рис. 10) содержит данные о моделировании полного теста, содержащего 8 входных наборов, с помощью сервисной процедуры RUN.

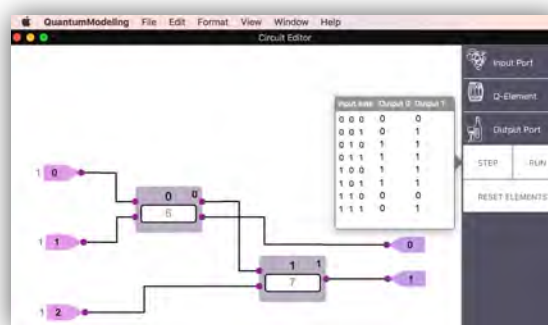


Рис. 10. Моделирование схемы на полном тесте

Для моделирования цифровой схемы вводятся кубитные покрытия в десятичном виде, что достаточно удобно для простейших логических элементов (рис 11). Чтобы наблюдать состояния выходных линий, можно воспользоваться соединением внутренних линий с выходными значками интерфейса. Это также дает возможность автоматически формировать таблицу моделирования на полном проверяющем тесте, содержащем 2^n входных последовательностей.

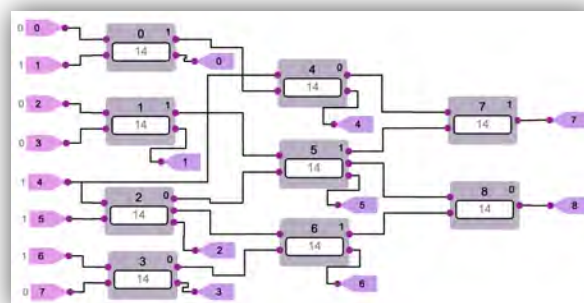


Рис. 11. Моделирование схемной структуры на тест-векторе

Реализация Хассе-процессора (рис. 12) для решения задачи покрытия содержит логические элементы, которые содержат одновходовые примитивы, двухвходовые, трехвходовые и один

четырёхходовой элемент. Практически за один автоматный такт осуществляется поиск оптимального покрытия, если элементы реализуют регистровые переменные по входам и выходам. Состояние ближайшего к внешним входам выхода логического элемента, равное единице, свидетельствует о наличии оптимального решения.

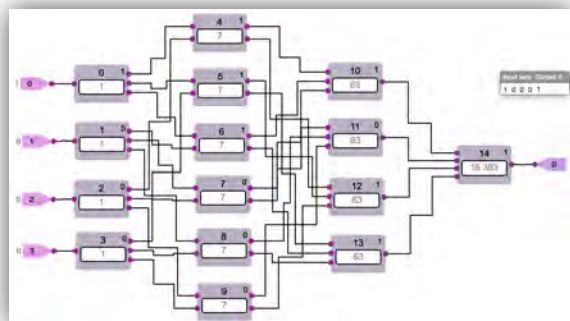


Рис. 12. Моделирование Хассе-структуры на тест-векторе

Таким образом, сервис логического моделирования способен продемонстрировать преимущества гибких кубитных структур данных и адресно-ориентированного анализа цифровых устройств, как альтернативной технологии для memory-driven квантового параллельного компьютеринга для решения задач синтеза и анализа.

Здесь следует также отметить необходимость введения в будущем: новой модели кубитных неисправностей, ориентированной на memory-driven компьютеринг; задания кубитного формата входных воздействий, аналогичного кубитным векторам логических элементов; синтеза кубитных моделей для цифровых автоматов, где уже не нужны триггеры для синтеза функций возбуждения; параллельного моделирования без наличия адресов памяти; моделирования неисправностей, привязанных к кубиту.

10. Облачная имплементация сервиса “Quantum modeling”

Реализация квантового моделирования в качестве облачного сервиса имеет ряд преимуществ перед программным приложением: 1) Инвариантность к аппаратным вычислительным средствам как к быстро устаревающей основе, на которой функционирует сервис. 2) Отсутствие необходимости в покупке аппаратных средств для реализации программного приложения как сервиса. 3) Низкий уровень финансовых затрат для аренды облачного пространства, как платформы для реализации компьютерингового сервиса. 4) Инвариантность места нахождения разработчика (-ов) для быстрой имплементации облачного сервиса. 5) Инвариантность облачного сервиса к географическим координатам всех

пользователей, находящихся на планете. 6) Доступность облачного сервиса с любой точки земного шара во времени и пространстве. 7) Высокий уровень надежности дата-центров, формирующих облачно-ориентированные платформы от компаний: Google, Amazon, Microsoft. 8) Высокий уровень software-driven кибербезопасности облачных сервисов, предоставляемых упомянутыми платформами. 9) Быстрая технологическая масштабируемость облачных сервисов, ориентированная на расширение функциональных возможностей и большое количество реальных пользователей.

Конкретно, облачный сервис “Quantum modeling” реализован на платформе Google, которая обеспечивает средства для реализации инфраструктурных микросервисов: хранения данных и доступа к ним, киберзащиты облачной функциональности и аутентификации пользователей. Инфраструктурные микросервисы от Google платформы существенно влияют на производительность и снижение времени проектирования облачных сервисов, а также на технологические удобства поддержания работающего сервиса в актуальном режиме функционирования.

Облачный сервис “Quantum modeling” представлен доменно и IP-идентифицируемой структурой киберфизических компонентов, реализующих компьютеринг-сервис моделирования цифровых устройств в контейнерно-ориентированной среде Docker, работающей в Google Computing Engine.

Цель сервиса связана с популяризацией новых технологий квантового синтеза и анализа цифровых устройств в среде студентов, ученых и специалистов, благодаря упомянутым выше преимуществам облачного компьютеринга.

Задачи: 1) Создание облачных микросервисов квантового проектирования, моделирования, диагностирования и верификации широкого класса цифровых устройств. 2) Популяризация кубитных структур данных и квантовых технологий проектирования memory-driven вычислительных устройств.

Современный рынок проектирования облачных систем показывает, что в последние три года набирают популярность гибкие и масштабируемые технологии проектирования сервисов, использующих контейнеризацию, направленную на минимизацию time-to-market. Основная идея контейнера заключается в создании защищенной и доступной виртуальной среды проектирования и эксплуатации в киберпространстве на основе использования локальной или глобальной платформы (рис. 13).

На инфраструктуру платформы устанавливается практически любая операционная система Host

operating system, удобная для разработчика, например, Ubuntu, Debian. В локально исполненной hardware инфраструктуре контейнеров можно установить также OSX и Windows.

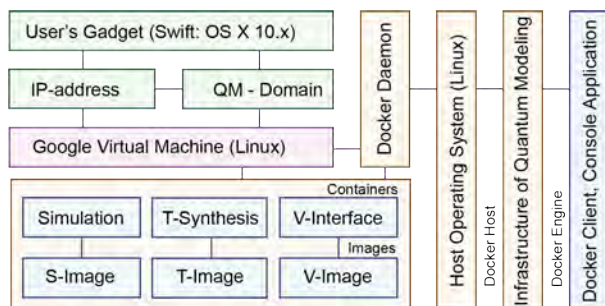


Рис. 13. Инфраструктура сервиса моделирования

Разработчик устанавливает Docker Engine, если он не установлен изначально, который с помощью Docker Daemon (реализация на Golang, C) управляет контейнерами, запущенными на основе предварительно построенных образов (Images). Контейнеры Containers представляют собой легковесную виртуальную вычислительную машину, под которой работают функциональные блоки: Modeling, Simulation, Test Synthesis, Fault Simulation, Fault Diagnosis, которые реализованы на языке Swift (Jawa) и имеют соответствующие библиотеки. Docker Engine представляет собой компьютерную систему облачной контейнеризации, состоящую из двух взаимодействующих между собой компонентов: Docker Host, Docker Client (Console Application), которые можно интерпретировать в качестве исполнительного Host и управляющего (Client) механизма облачной системы Quantum Modeling. Далее Docker Engine является компонентом, входящим в состав Google Virtual Machine, которой ставится в соответствие IP-address и Quantum Modeling domain, что делает разработку, а далее облачный сервис, доступным для широкого круга специалистов и студентов.

Последним звеном в вертикальной связке (Back-End, Front-End) является User's Gadget (Swift OS X.10x). На конечном устройстве пользователя устанавливаются Front-End приложения, способные привести результаты работы облачных сервисов моделирования к мультиоконному формату визуализации данных, удобных для восприятия человеком. Следует отметить, что намного технологичнее создать Front-End в виде приложения на JavaScript, HTML, CSS, которое будет работать через Internet Browser. Это гарантирует отсутствие блокирования клиента со стороны поставщика конкретной ОС (vendor lock). Для мобильных устройств целесообразнее использовать приложение индивидуального клиента под

каждую ОС. Все модули облачного сервиса запрограммированы на языке Swift, операционная система OSX 10.9, компилятор XCode 7. Количество исходных файлов 42, общее число строк кода – 2300.

Опытная эксплуатация Quantum Modeling сервиса показала: 1) Слушатели достаточно легко переходят на кубитный формат описания логических элементов в цифровой структуре. 2) Существенно уменьшается время проектирования цифровых устройств при использовании удобного графического интерфейса. 3) Значительно проще решаются все задачи, связанные с тестированием, моделированием и диагностированием цифровых компонентов и схем при применении кубитных структур данных и квантовых методов синтеза и анализа.

Для получения статистической информации и верификации Quantum Modeling было проведено 10 экспериментов на логических схемах из библиотеки ISCAS, а также на других структурах, приведенных ниже: 1) Adder SP. 2) Circuit Schneider. 3) Circuit C5. 4) Circuit C17. 5) RFO Circuit. 6) MUX16 Circuit. 7) DFA Circuit. 8) Hasse processor. 9) DC4-16 Circuit. 10) Circuit C432. Сравнению подлежало время ввода схемной структуры (Modeling Time), а также время проектирования диагностической информации (Designing Time), которая включает: ввод модели устройства, синтез и анализ тестов для получения таблицы неисправностей. Базовым вариантом для сравнения послужил продукт Active HDL, Aldec Inc., где информация о модели схемы вводилась на VHDL языке описания аппаратуры. Статистика сравнения двух времен: Modeling Time и Designing Time представлена на рис. 14 и 15 соответственно.

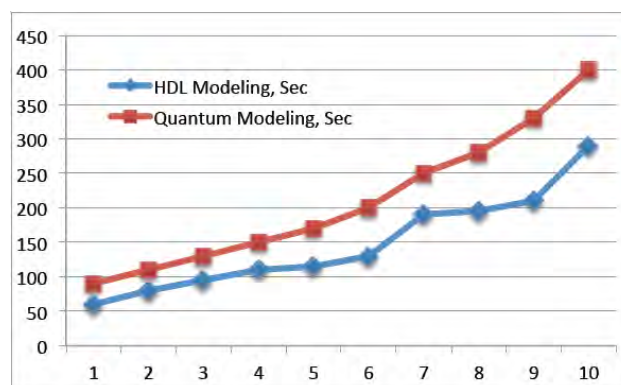


Рис. 14. Сравнительный анализ времени ввода логических схем

Она показывает преимущества визуального образного схемотехнического проектирования логических схем небольшой размерности по сравнению с вводом схем на основе HDL-описания. Особенно это приемлемо и эргономично в про-

цессе обучения студентов технологиям проектирования и верификации цифровых систем и компонентов. Однако для больших промышленных проектов пока еще целесообразно использовать языки описания аппаратуры.

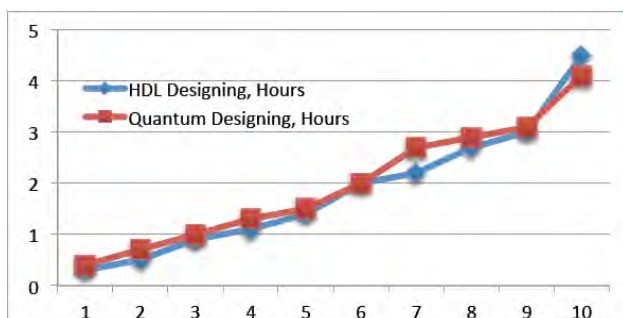


Рис. 15. Сравнительный анализ времени проектирования логических схем

Выводы. Становление новой теории проходит через ее первоначальное отторжение традиционно ориентированной научной общественностью. Тем не менее, очевидные отдельные преимущества практически направленного квантового компьютеринга находят отражение в публикациях ученых, работающих в областях, связанных с параллельными memory-driven вычислениями и комбинаторикой задач защиты информации, верификации и диагностирования. Нет лучшего способа продвинуть инновационную теорию путем использования открытого киберфизического пространства, где создаются облачные сервисы, покрывающие интересы широкой научной общественности во времени и пространстве. Поэтому соединение квантового компьютеринга с его облачным исполнением является удачной технологической связкой для решения задач киберфизического формата, ориентированного на проектирование и тестирование цифровых систем и программно-аппаратных компонентов. В рамках обозначенного научно-технологического направления были созданы следующие средства: 1) Графический интерфейс, удобный для ручного проектирования кубитных моделей цифровых устройств и компонентов, который дает возможность в режиме online осуществлять коррекцию ошибок. При высоком быстродействии современного компьютеринга главным фактором, влияющим на time-to-market, становится время ручного ввода моделей цифровых систем и компонентов. 2) Структуры данных для кубитного описания цифровых устройств и компонентов, которые отличаются компактностью и высоким параллелизмом их обработки. 3) Инфраструктура для проектирования и тестирования цифровых устройств и компонентов с функциями хранения, удаления и коррекции данных, что дает возможность осуществлять одновременное создание

цифровых схем несколькими проектировщиками. 4) Программные модули для квантового моделирования цифровых устройств и компонентов в режимах ручного и автоматического ввода входных тестовых последовательностей, что дает возможность наглядного обучения студентов методам синтеза и анализа. 5) Программные средства для синтеза таблиц неисправностей на основе дедуктивного метода квантового моделирования дефектов путем использования кубитных структур данных, которые обеспечивают поиск одиночных и кратных константных дефектов в режиме реального времени. 6) Направления будущих исследований связаны с созданием memory-driven моделей неисправностей, ориентированных на кубитные формы описания функциональностей с последующей разработкой семейства memory-driven параллельных методов квантового тестирования, моделирования и диагностирования компьютеринговых систем и компонентов.

11. Заключение

1. Кибертенденции от Gartner Inc. предоставляют возможность лидерам корпоративной архитектуры и руководителям университетов не отставать безнадежно от цифрового бизнеса в науке, образовании и индустрии, своевременно реагировать на киберфизические угрозы, возглавлять бизнес-инновации и определять эффективную цифровую бизнес-стратегию устойчивого развития государств.
2. По факту Нуре-cycle представляет собой глубокую 4D-аналитику, во времени и в пространстве, состояния современного рынка устойчивого киберфизического развития умных облачных технологий на ближайшие 10-15 лет.
3. Для университетов Нуре-цикл определяет жизненную необходимость инвестировать в знания студентов инновационные технологии, показанные в фазах цикла, в целях получения через 5-10 лет армии креативных специалистов, способных поднять государство из руин современного киберневежества. Иначе, Gartner цикл для университета представляет собой стратегию его киберфизического устойчивого развития во времени и пространстве. Любая стратегия, разработанная без знания темпов и направления технологических изменений, будет страдать неправильным планированием действий, разрушением бизнеса, науки и образования. Например, следует учитывать, что в 2018 году gobobossy будут точно мониторить и дистанционно online управлять 3 миллионами работников в мире с целью: метрически оценивать потенциал исполнителей, раздавать задания, логистически верно маршрутизировать их успешное выполнение, инвариантное к позиционированию работника в физическом

пространстве, оценивать качество и производительность труда, начислять заработную плату по метрическим результатам.

4. Нуре-цикл неявно дифференцирует все топ-технологии на ведущие и ведомые (master-slave), которые по факту означают, что развитию HardWare (Physical Space) платформ в сторону компактности всегда отдается приоритет, поскольку остальной виртуальный мир (Cyber Space), стремящийся к безграничному расширению SoftWare приложений, всегда будет ведомым. Взаимодействие двух миров, связанных с устойчивым развитием объемов аппаратного и программного обеспечения, формирующего киберфизическое пространство, представлено на рис. 16.

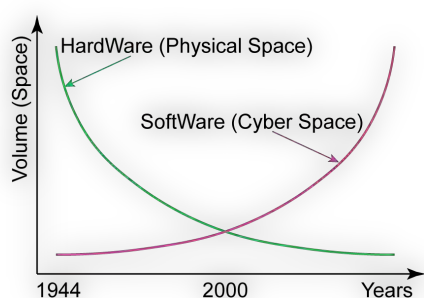


Рис. 16. Взаимодействие объемов киберфизических компонентов

5. Тем не менее, аппаратные и программные технологии представлены в Нуре-цикле (на рынке) практически в одинаковых пропорциях (50:50): Hardware-driven technologies: 4D Printing, Volumetric Displays, Nanotube Electronics, Brain-Computer Interface, Human Augmentation, Autonomous Vehicles, Cognitive Computing, Commercial UAVs (Drones), Smart Dust, Smart Robots, Smart Workspace, Connected Home, 5G, IoT Platform, Edge Computing, Neuromorphic Hardware, Quantum Computing;

Software-driven technologies: Deep Learning, Deep Reinforcement Learning, Artificial General Intelligence, Enterprise Taxonomy, Ontology Management, Machine Learning, Virtual Assistants, Cognitive Expert Advisors, Digital Twin, Blockchain, Serverless PaaS, Software-Defined Security, Virtual Reality, Augmented Reality, Augmented Data Discovery, Conversational User Interfaces, Digital Humanity, Smart Cyber Digital State.

6. Одинаковое соотношение аппаратных и программных технологий в Gartner-прогнозе означает, что уровни их капитализации на NASDAQ-рынке стремятся к паритету, ярким примером которого являются компании Apple (800 млрд долларов – индекс NASDAQ 2017) и Google (570 млрд). Эти производители существенно отличаются тем, что они полагаются на мудрость своих команд (экспертов), вооруженных доктриной:

«потребители не могут предсказывать свои собственные потребности» (consumers could not predict their own needs) [9]. Альтернативой является политика компании Microsoft (503 млрд), которая проводит обширные исследования перед запуском продукта, например, такого как Windows Phone. По оценкам Gartner доля Apple на мировом рынке мобильных телефонов составляет 14,2% против 3,3% для Microsoft. Кому доверять, экспертам или потребителям? Ответ однозначный – экспертам, в формате 4D (всегда, везде и по всем вопросам).

7. Представлена memory-driven инновационная архитектура квантового компьютеринга, которая определяется возможностью устранения логики, связанной с суперпозицией и перепутыванием состояний на основе использования характеристического уравнения, реализующего транзакции записи-считывания на структуре электронов. Исключение логических операций из квантового компьютеринга позволяет существенно упростить архитектуру до уровня структуры памяти на электронах для выполнения транзакций между ними с помощью квантов или фотонов. Показано формальное отличие квантового компьютеринга от классического, которое заключается в возможности параллельного и одновременного выполнения логических операций над множествами.

8. Предложен квантовый метод минимизации булевых функций, который отличается от метода неопределенных коэффициентов параллельным выполнением операции суперпозиции над нулевыми и единичными состояниями входных переменных, представленными унитарными кодами, что дает возможность существенно повысить быстродействие за счет избыточной памяти.

9. Предложен кубитный метод безусловного диагностирования неисправностей цифровых устройств, который отличается от аналогов параллельным выполнением логических операций, что дает возможность существенно повысить быстродействие поиска дефектов за счет унитарного кодирования данных.

10. Представлены кубитные структуры данных, облачная реализация методов и моделей анализа, что дает возможность изучать квантовые технологии тестирования и моделирования цифровых систем и компонентов в режиме online.

Литература:

1. <https://www.forbes.com/sites/louisacolumbus/2017/08/15/gartners-hype-cycle-for-emerging-technologies-2017-adds-5g-and-deep-learning-for-first-time/#646a4cf34be2>
2. <http://www.gartner.com/newsroom/id/3784363>
3. <http://www.wired.co.uk/article/ai-neuromorphic-chips-brains>

4. A. Gupta and R. K. Jha, "A Survey of 5G Network: Architecture and Emerging Technologies," in IEEE Access, vol. 3, P. 1206-1232, 2015.
5. C. Zhu, V. C. M. Leung, L. Shu and E. C. H. Ngai, "Green Internet of Things for Smart World," in IEEE Access, vol. 3. P. 2151-2162, 2015.
6. K. Christidis and M. Devetsikiotis, "Blockchains and Smart Contracts for the Internet of Things," in IEEE Access, vol. 4. P. 2292-2303, 2016.
7. Blockchains: How They Work and Why They'll Change the World IEEE Spectrum. October 2017. <https://spectrum.ieee.org/computing/networks/blockchain-s-how-they-work-and-why-theyll-change-the-world>
8. A. Zanella, N. Bui, A. Castellani, L. Vangelista and M. Zorzi, "Internet of Things for Smart Cities," in IEEE IoT Journal, vol. 1, no. 1. P. 22-32, Feb. 2014.
9. https://www.gartner.com/doc/3471559?srcId=1-7578984202&utm_campaign=RM_GB_2017_TRENDS_QC_E2_What&utm_medium=email&utm_source=Eloqua&cm_mmc=Eloqua_-Email_-LM_RM_GB_2017_TRENDS_QC_E2_What_-0000
10. <http://www.gartner.com/smarterwithgartner/three-digital-marketing-habits-to-break-2/>
11. Vladimir Hahanov. Cyber Physical Computing for IoT-driven Services. New York. Springer. 2017. 243p.
12. Hahanov V.I., Bani Amer Tamer, Chumachenko S.V., Litvinova E.I. (2015) Qubit technology for analysis and diagnosis of digital devices. Electronic modeling. J 37 (3): 17-40.
13. Vladimir Hahanov, Wajeb Gharibi, Eugenia Litvinova, Mykhailo Liubarskyi, Anastasia Hahanova. Quantum memory-driven computing for test synthesis. IEEE East-West Design and Test Symposium. 2017. Novi Sad, Serbia. P. 123-128.
14. Tian Zhong¹, Jonathan M. Kindem¹, John G. Bartholomew¹ at all. Nanophotonic rare-earth quantum memory with optically controlled retrieval. Science. 29 Sep 2017: Vol. 357, Issue 6358, pp. 1392-1395.
15. Jungkil Kim, Hoo-Cheol Lee, Kyoung-Ho Kim at all. Photon-triggered nanowire transistors. Nature Nanotechnology 12. P. 963–968 (2017).
16. Giacomo Lovat, Bonnie Choi, Daniel W. Paley at all. Room-temperature current blockade in atomically defined single-cluster junctions. Nature Nanotechnology, doi:10.1038/nnano.2017.156.
17. Chao Li, Zhongping Wang, Yan Lu, Xiaoqing Liu & Li Wang. Conformation-based signal transfer and processing at the single-molecule level. Nature Nanotechnology. (2017), doi:10.1038/nnano.2017. 179.
18. Patrick T. Hester, Kevin Adams. Systemic Decision Making. Fundamentals for Addressing Problems and Messes. Springer Nature. 412 p.
19. Hahanov Vladimir, Barkalov Alexander, Adamski Marian. Infrastructure intellectual property for SoC simulation and diagnosis service. Springer, Germany. 2011.– P. 289-330.
20. <http://dpchas.com.ua/politika/dubilet-zadumal-revoluciyu>
21. Савельев А.Я. Прикладная теория цифровых автоматов. М.: Высш. шк., 1987. 272 с.
8. V. Kharchenko, Y. Kondratenko, J. Kacprzyk (Eds.), "Green IT Engineering: Concepts, Models, Complex Systems Architectures", In the book series "Studies in Systems, Decision and Control" (SSDC), vol. 1, Berlin, Heidelberg: Springer International Publishing, 2017.
9. V. Kharchenko, Y. Kondratenko, J. Kacprzyk (Eds.), "Green IT Engineering: Components, Networks and Systems Implementation". In the book series "Studies in Systems, Decision and Control" (SSDC), vol. 2, Berlin, Heidelberg: Springer International Publishing, 2017.

Поступила в редколлегию 12.12.2017

Рецензент: д-р техн. наук, проф. Кривуля Г.Ф.

Емельянов Игорь Валериевич, научный сотрудник кафедры АПБТ ХНУРЭ. Научные интересы: проектирование и тестирование цифровых систем. Хобби: путешествия. Адрес: Украина, 61166, Харьков, пр. Науки, 14, e-mail: iyemelyanov@itdelight.com.

Любарский Михаил Михайлович, соискатель кафедры АПБТ ХНУРЭ. Научные интересы: проектирование и тестирование цифровых систем. Хобби: путешествия. Адрес: Украина, 61166, Харьков, пр. Науки, 14.

Хаханов Владимир Иванович, д-р техн. наук, проф., главный научный сотрудник кафедры АПБТ ХНУРЭ. Научные интересы: проектирование и тестирование цифровых систем. Хобби: футбол, горные лыжи. Адрес: Украина, 61166, Харьков, пр. Науки, 14, e-mail: hahanov@icloud.com.

Iemelianov Igor Valerievich, Research Associate, Design Automation Department, NURE. Scientific interests: design and testing of digital systems. Hobbies: traveling. Address: Ukraine, 61166, Kharkov, Nauki Ave, 14, e-mail: iyemelyanov@itdelight.com.

Lyubarsky Mikhail Mikhailovich, PhD student, Design Automation Department, NURE. Scientific interests: project-bathing and testing digital systems. Scientific interests: design and testing of digital systems. Hobbies: traveling. Address: Ukraine, 61166, Kharkov, Nauki Ave, 14.

Hahanov Vladimir Ivanovich, Dr. of Tech. Sc., Prof., Chief Scientific Officer, Design Automation Department, NURE. Scientific interests: design and testing of digital systems. Hobby: football, downhill skiing. Address: Ukraine, 61166, Kharkov, Science, 14, e-mail: hahanov@icloud.com.

МУЛЬТИВЕРСНЫЙ ПАРАЛЛЕЛЬНЫЙ СИНТЕЗ ЦИФРОВЫХ СТРУКТУР НА ОСНОВЕ SYSTEMC СПЕЦИФИКАЦИИ

ОБРИЗАН В.И., СОКЛАКОВА Т.Г.

Обобщаются исследования, связанные с мультиверсным параллельным синтезом цифровых структур на основе SystemC спецификации, цель которого – существенное уменьшение времени проектирования вычислительных архитектур и повышение качества цифровых изделий.

Keywords – System-C, VHDL, Verilog, киберсистема, синтез.

1. Введение

Создание цифровых систем на кристаллах играет ключевую роль для развития компьютерных технологий, используемых в экономике, социологии, медицине, бизнесе, промышленности, образовании, которые формируют киберэкосистему планеты: распределенные центры больших данных, Internet of Things, Internet of Everything, cyber physical system, cloud-mobile computing, service computing.

Важно отметить экспоненциальный рост не только производительности компьютерных систем, но и продуктивности инженеров, благодаря созданию облачных сервисов проектирования аппаратных и программных приложений, интегрированных в понятие компьютинг (рис. 1).

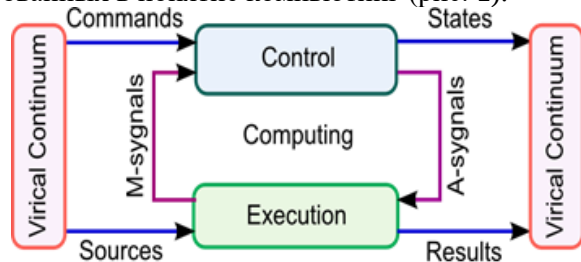


Рис. 1. Компьютинг для создания цифровых систем на кристалле

Компьютинг — процесс достижения поставленной цели путем использования ресурсов, механизмов управления и исполнения в циклически замкнутой системе с заданными отношениями и результирующей продукцией, сигналами мониторинга и актуации. Развитие облачных сервисов не имеет целью сократить рабочие места, а предоставить инженерам инструментарий для выполнения высокотехнологических работ в любой точке земного шара по созданию качественных программно-аппаратных приложений, минимизирующих время выхода годной продукции на рынок. Технологии облачных сервисов по разработке цифровых систем на кристаллах системного уровня позволяют создавать исполняемые спецификации — описание проекта на функциональном уровне, которое можно верифицировать в удаленном online-режиме. На функциональном

уровне спецификация отражает вопросы «что делать?», а не «как делать», таким образом, она не содержит требований к структуре или архитектуре целевой платформы. При этом компонентами создания некоторой цифровой системы выступают уже сервисы, а не функциональные элементы. Такая замена дает существенный выигрыш во времени проектирования, которое определяется быстродействием процедур верификации и тестирования за счет варьирования параметрами системного, RT и вентиляционного уровней, добываясь лучших характеристик реализации цифрового изделия на кристалле.

Цель — существенное уменьшение времени проектирования вычислительных архитектур и повышение качества цифровых изделий путем мультиверсного синтеза структуры цифрового изделия на основе заданной спецификации в среде SystemC (C++) и автоматического подбора функциональных компонентов за счет параллельного синтеза и верификации архитектурных решений системного уровня в соответствии с предложенной метрикой.

2. Методы системного и архитектурного синтеза интерфейсных структур

Модель системного уровня может не иметь однозначного интерфейса на логическом уровне. Разработанный метод определяет интерфейс модели, чтобы ее можно было однозначно представить на логическом уровне. Метод рассматривает прототипы C++-функций: количество аргументов, их типы и типы возвращаемых значений. В разделе представлено соответствие стандартных C++-типов типам языка VHDL. Следующий метод синтезирует протокол взаимодействия с модулем на логическом уровне. Для выбранного протокола взаимодействия строится граф-схема алгоритма, которая объединяется с граф-схемой проектируемого устройства. Таким образом, синтезируемое устройство сопряжено с другими устройствами, работающими на той же самой шине данных.

Обычно высокоуровневые модели не имеют понятия модельного времени: все вычисления происходят на одном условном такте работы. Модели же на логическом уровне работают по тактам. Таким образом, можно выделить две подзадачи: а) определение соответствия аргументов функции на системном уровне и сигналов на логическом уровне; б) определение протокола взаимодействия с синтезируемой функцией. Здесь протокол — это последовательность изменения информационных и управляющих сигналов схемы, которая приводит к выполнению синтезируемой функции.

Рассмотрим синтез интерфейсов для популярного стандарта шины системы на кристалле Wishbone.

Этот стандарт предполагает различные сопряжения модулей в системе: соединение «точка-точка»; конвейерное соединение (для обработки потока данных); соединение «многие ко многим» через общую шину; соединение «многие ко многим» через коммутатор; соединение в составе сети на кристалле.

Wishbone предоставляет шину данных, размер которой может быть от 8 до 64 бит, но всегда кратный одному байту. Таким образом, для передачи параметров используется последовательная загрузка значений во внутреннюю память ведомого модуля. Для этого используется специальная адресация регистров, которые расположены в интерфейсе ведомого модуля.

В том случае, если размер шины меньше 32 бит, требуется несколько циклов чтения результата из регистра R1. Можно рассчитать количество тактов, необходимых для передачи параметров. Пусть P_i – вектор параметров функции, $i = 1, \dots, n$, где n – количество параметров, передаваемых в функцию; $S(P_i)$ – размер параметра P_i в байтах; M – разрядность шины данных в байтах.

На рис. 2 показаны алгоритмы работы ведущего и ведомого интерфейсов в системе Wishbone. В состоянии a_3 показан вызов функции $f1$ с параметром par . Согласно этой ГСА вычисление функции занимает ровно один такт работы автомата. Это может быть справедливо лишь для высокоуровневых моделей без отсутствия модельного времени. В системах на структурном уровне описания такого можно добиться только для комбинационных функций. Для автоматов, требующих большее количество состояний для завершения, необходимо изменить ГСА (а) следующим образом. На месте состояния a_3 необходимо подставить ГСА вызываемой функции $f1$.

3. Модели описания логических блоков системного уровня

На первом этапе разработки модели контейнера «вектор» системного уровня был формализован интерфейс SystemC-модуля для контейнера «вектор». На рис. 3 показана система, включающая все необходимые компоненты системного уровня. Clock — это генератор тактовых импульсов. Контейнер «вектор» — синхронное устройство, у которого все операции осуществляются по переднему фронту сигнала синхронизации.

Генератор тестов — это высокоуровневый модуль, который содержит генератор тестов для контейнера и анализатор ответных реакций. Этот генератор будет повторно использован для верификации модели на логическом уровне: H2L (high level to low level) – транзактор, который преобразует высокоуровневые вызовы функций контейнера к протоколу на уровне переключения логических сигналов; L2H (low level to high level) – транзактор, который осуществляет преобразование, обратное H2L – протокол уровня логических сигналов к высокоуровневым вызовам функций

контейнера «вектор». На рис. 4 показан подробный интерфейс транзактора L2H. Этот же интерфейс будет иметь будущая аппаратная реализация.

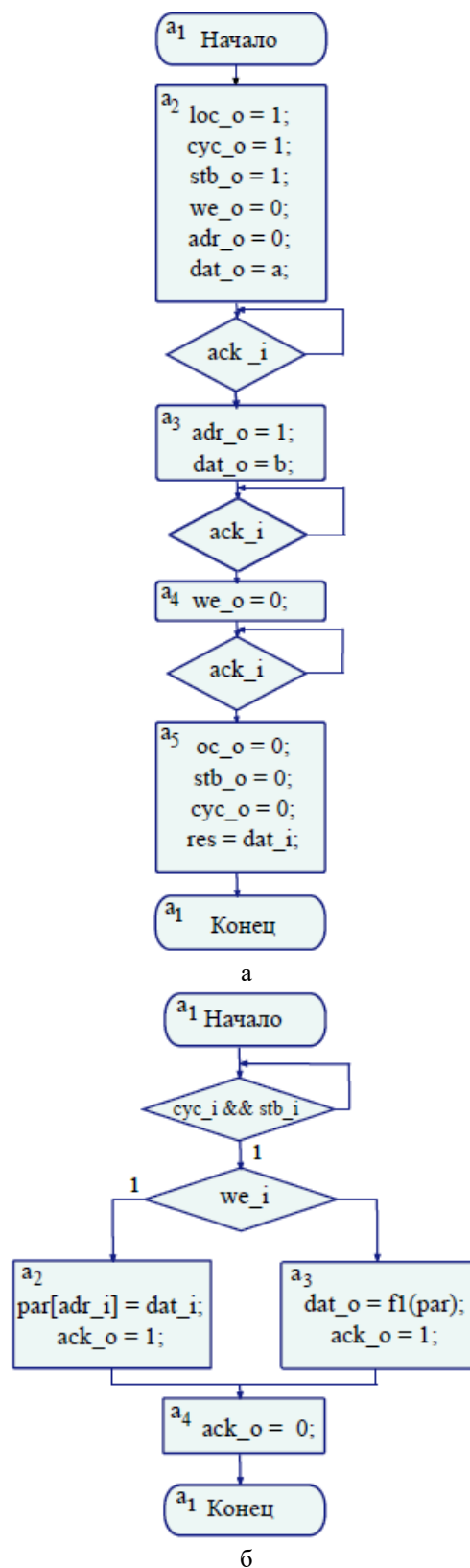


Рис. 2. Алгоритмы ведущего и ведомого интерфейсов в системе Wishbone

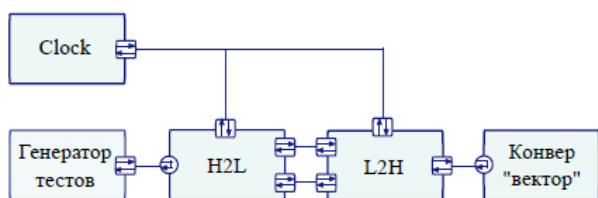


Рис. 3. Полный комплект модулей и транзакторов для разработки системной модели контейнера «вектор»

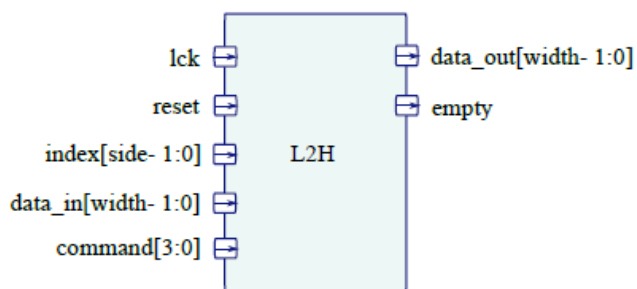


Рис. 4. Интерфейс логического уровня модели контейнера «вектор»

Логическая модель контейнера «вектор» реализована с помощью примитивов блочной памяти кристалла Xilinx Virtex-5. Один блок памяти кристалла рассчитан на 36 кбит данных и может быть сконфигурирован или как два независимых модуля памяти по 18 кбит, или как один модуль емкостью 36 кбит. Можно пользоваться следующими конфигурациями памяти: 32K по одному биту, 16K по 2, 8K по 4, 4K по 9, 2K по 18 или 1K по 36 бит.

Выбор кристаллов Xilinx Virtex определяется следующими факторами. Эти кристаллы доступны на рынке в низких партиях за приемлемые цены. Здесь они существенно выигрывают перед кристаллами жесткой логики, для которых только подготовительные затраты на изготовление маски и пробную серию могут достигать миллионов долларов. ПЛИС Virtex также имеет возможность многократного перепрограммирования, что позволяет испытывать различные конфигурации проектируемого устройства, а также исправлять логические дефекты в проекте. Кристаллы жесткой логики такой возможности не имеют.

Кристаллы Xilinx Virtex также имеют необходимые ресурсы для реализации контейнера «вектор»: элементы блочной памяти. Блочная память поддерживает синхронные операции записи/чтения, два порта доступа полностью симметричны и работают независимо друг от друга, разделяя данные в памяти.

Ограничения логической модели: контейнер занимает всю память, которую предоставляет базовый элемент блочной памяти, использованный в реализации; максимальное количество элементов вектора определяется размером одного элемента и объемом элементов памяти, использованным в реализации. Может быть вычислено по формуле

$V = w \times k$, где w — размер элемента в битах, k — количество элементов, V — объем занимаемой памяти.

Для успешной реализации модели контейнера «вектор» была разработана дополнительная логика, реализующая требуемые операции для элементов памяти.

Контейнер «вектор» предоставляет полный набор конструкторов, деструкторов и операций копирования. Для реализации множества операций используется дешифратор команд, который переводит из двоичного кода инструкции в позиционный код, которые активирует управляющие сигналы контейнера. Логическая модель имеет следующие параметры: DATA_WIDTH определяет ширину шины данных. Этот параметр зависит от того, для какого типа данных создан контейнер «вектор». Параметр выбирается в зависимости от максимального количества доступных команд, число которых равно 2^{CMD_WIDTH} ; SIZE определяет ширину шины адреса. Этот параметр влияет на максимальное количество адресуемых ячеек контейнера и равно 2^{SIZE} .

Логическая модель контейнера «вектор» имеет интерфейсные сигналы: глобальный синхроимпульс *clk*; сброс *reset*, при высоком уровне которого происходит сброс элементов памяти в нули, что соответствует очистке контейнера; шина адреса *index* размерностью [SIZE-1:0], которая используется для указания порядкового номера элемента в векторе при записи или чтении; шина данных *data_in* размерностью [DATA_WIDTH-1:0], которая используется для записи данных в вектор; шина команд *command* размерностью [CMD_WIDTH-1:0], которая используется для управления контейнером; шина данных *data_out* размерностью [DATA_WIDTH-1:0], которая используется для чтения данных из вектора; сигнал *empty*, высокий уровень которого сигнализирует о том, что вектор пуст.

Структурная модель контейнера «вектор» содержит дешифратор команд *dc*, который преобразует двоичный код в управляющие сигналы.

Блок *pointer* содержит два регистра: адрес последнего элемента (для пустого вектора он равен - 1) и адрес первого свободного элемента (для пустого вектора равен нулю). На входе блока есть три управляющих сигнала: *reset*, *inc*, *dec*. При высоком уровне сигнала *inc* и переднем фронте синхроимпульса внутренние регистры увеличиваются на единицу. При высоком уровне сигнала *dec* и переднем фронте синхроимпульса внутренние регистры уменьшаются на единицу.

Мультиплексор индекса управляется шиной из двух сигналов. В режиме чтения/записи ячейки по индексу сигналы должны быть установлены в 00, в режиме *push_back()* сигналы должны быть установлены в 01, в режиме *back()* сигналы должны быть установлены в 10, в режиме *front()* сигналы должны быть установлены в 11.

Для анализа и проектирования контейнера «список» рассмотрим типичные операции, выполняемые контейнером «вектор». Здесь X – это условие, возбуждающее ту или иную операцию, команды и описание команд: $x_1_begin()$ – ссылка на первый элемент списка; $x_2_end()$ – ссылка на последний элемент списка; $x_bool_empty()$ – возвращает true, если список пуст; $x_3_push_front()$ – добавляет новый элемент в начало; $x_4_push_back()$ – добавляет новый элемент в конец; $x_5_pop_front()$ – удаляет элемент из начала списка; $x_6_pop_back()$ – удаляет элемент из конца списка; $x_7_insert()$ – добавляет элемент перед текущим; $x_8_erase()$ – удаляет текущий элемент; $x_9_next()$ – ссылка на следующий элемент; $x_{10_prev}()$ – ссылка на предыдущий элемент.

Для успешной реализации модели контейнера «список» была разработана дополнительная логика, реализующая требуемые операции.

Для верификации полученных моделей были использованы тесты, проверяющие 100% режимов работы устройства. Исходные эталоны были получены из программы на C++.

Здесь рассматриваются только функциональные тесты – проверка правильности функционирования модели, и нагрузочные тесты – замеры производительности и временных характеристик модели. На рис. 5 показана схема процесса верификации логической модели.

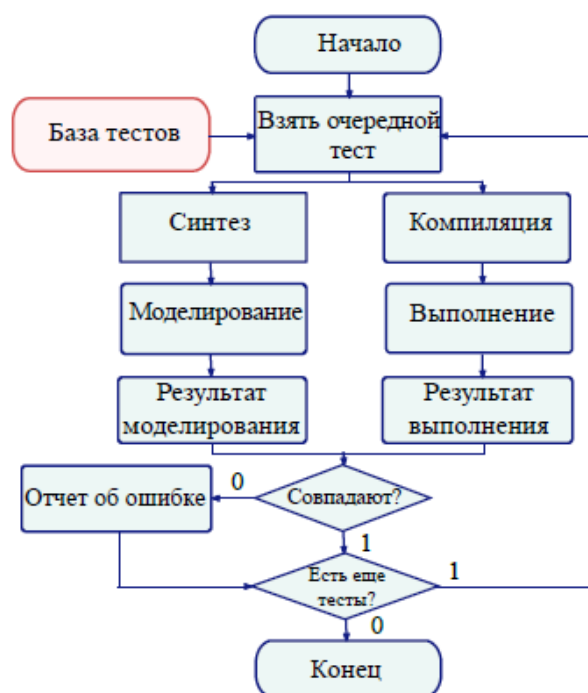


Рис. 5. Схема верификации логической модели

Создается набор тестов на C++, в которых используются контейнер «вектор». В этих тестах присутствуют вызовы всех методов, которые поддерживает контейнер. На следующем этапе тесты компилируются обычным C++ компилятором для получения исполнимого файла. При исполнении

этой программы в файле сохраняются результаты работы контейнера.

С другой стороны, происходит синтез исходных текстов. В результате синтеза получается модель системного уровня, содержащая контейнер «вектор», описанный на ЯОА VHDL. Далее осуществляется моделирование с помощью программы Aldec Active-HDL для получения результатов. Эти результаты сравниваются с результатами, полученными при работе обычной программы. Логическая модель составлена верно, если результаты моделирования и выполнения совпадают.

Следующим этапом верификации является проверка соответствия функциональной модели и модели с временными задержками. В функциональной модели отсутствует информация о времени распространения сигнала от входов элементов к выходам. Подразумевается, что сигнал распространяется мгновенно или за один дельта-цикл моделирования. Программа логического синтеза, трассировки и размещения генерирует специальный файл задержек распространения сигнала для выбранной архитектуры. Этот файл может быть загружен в САПР Aldec Active-HDL для построения временной модели.

Последним этапом тестирования является проверка модели, реализованной в том или ином кристалле. В работе выбрана архитектура ПЛИС Xilinx – дешевая и высокопроизводительная микросхема с возможностью многократного перепрограммирования.

4. Программно-аппаратная реализация моделей

Представлен метод мультиверсного синтеза управляющих и операционных автоматов в заданной инфраструктуре проектирования, ориентированных на архитектурные решения в метрике, минимизирующий время выполнения функциональности за счет распараллеливания операций при ограничении на аппаратные затраты.

По своей сути программа синтеза C++ в VHDL является машиной по трансформации исходного описания в результирующее. Таких трансформаций происходит несколько. Входная модель представлена на языке C++. Это алгоритмическое описание, которое решает поставленную перед инженером задачу.

Выполнено тестирование и верификация программных модулей инфраструктуры проектирования цифровых систем на кристаллах, а также определение эффективности предложенных моделей, методов и структур данных при создании реальных компонентов цифровых изделий.

Тестирование системы состоит из нескольких этапов: разработка схемы тестирования компилятора; подготовка тестов; подготовка эталонов; исполнение тестов и анализ результатов. В работе [11] описаны этапы преобразования.

Проведено сравнение производительности разработанной аппаратной модели и программной реализации на микропроцессоре экспериментальным путем. Данные получены вследствие работы 10 инженеров над одним проектом. Оценивались следующие характеристики: временные затраты на проектирование, быстродействие, энергопотребление, площадь на кристалле. В результате исследования получили при аддитивной оценке автоматический метод лучше в 1,301292012 раз (рис. 6).



Рис. 6. Аддитивная оценка эффективности

5. Заключение

Сущность рыночно-ориентированного научно-технического исследования определена как мультиверсное проектирование архитектуры цифрового изделия на основе заданной спецификации в среде SystemC (Си++) и автоматического выбора синтезируемых функциональных структур в целях существенного уменьшения времени создания проекта и повышения выхода годной продукции за счет параллельного синтеза и верификации архитектурных решений системного уровня в соответствии с предложенной метрикой. Основная инновационная идея заключается в параллельном автоматическом синтезе квазиоптимальной архитектуры в соответствии с предложенной спецификацией и метрикой на основе подбора синтезируемых функциональных структур. Достигнута цель исследования как существенное уменьшение времени проектирования вычислительных архитектур и повышение качества цифровых изделий.

Литература:

1. Hahanov V., Yemelyanov I., Obrizan V., Hahanov V. "Quantum" Diagnosis and Simulation of SoC // Proceedings of XIth Intern. Conf. MEMSTECH. 2015. P. 58-60.
2. Vladimir Obrizan, Igor Yemelyanov, Vladimir Hahanov, Eugeniya Litvinova. Matrix-Model for Diagnosing SoC HDL-Code // Radioelektronika and informatics. 2013. №1. P.12-19.
3. Hahanova, I.V.; Obrizan, V.; Adamov, A.; Shcherbin, D. Transaction level model of embedded processor for vector-logical analysis //, 2013 East-West Design & Test Symposium, pp.1-4, 27-30 Sept. 2013.

4. Obrizan, V. A method for automatic generation of an RTL-interface from a C++ description // East-West Design & Test Symposium (EWDTS). 17-20 Sept. 2010. pp. 186-189.
5. Volodymyr Obrizan. A Method of High-Level Synthesis and Verification with SystemC Language // Радиоэлектроника и информатика. 2010. №4.
6. Technologies for hardware simulation and verification / V. Hahanov, A. Hahanova, V. Obrizan, K. Zaharov // Proc. of the International Conference Modern Problems of Radio Engineering, Telecommunications and Computer Science – TCSET'2008, February 19–23. Slavske, Lviv, Ukraine. 2008. P. 560–564.
7. Testing challenges of SoC hardware-software components / V. Hahanov, V. Obrizan, S. Miroshnichenko, A. Gorobets // Proceedings of IEEE East-West Design and Test International Symposium, October 9–12, 2008. Lviv, Ukraine. P. 149–154
8. Hardware Simulation and Verification Technologies / V. Hahanov, A. Hahanova, V. Obrizan, W. Ghribi // Proceedings of IEEE East-West Design and Test Symposium, September 7–10, 2007, Yerevan, Armenia. P. 739–744.
9. Parallel Logic Simulation Using Multi-Core Workstations / V. Hahanov, V. Obrizan, A. Gavrushenko, S. Mikhtonyuk // The Experience of Designing and Application of CAD Systems in Microelectronics: proceedings of the 9th International Conference, February 19–24, 2007, Polyana, Ukraine. Lviv, 2007. P. 256–257.
10. Hierarchical Testing of Complex Digital Systems / V. Hahanov, V. Obrizan, V. Yeliseev, W. Ghribi // Proc. Of the Modern Problems of Radio Engineering, Telecommunications and Computer Science – TCSET'2006. February 28 – March 4, 2006, Slavske, Lviv, Ukraine. Lviv, 2006. P. 426–429.
11. Обризан В.И. Инфраструктура проектирования SOC для метода мультиверсного синтеза / В.И. Обризан // Радиоэлектроника и информатика. 2016. №2. С. 48-60.

Поступила в редколлегию 17.12.2017

Рецензент: д-р техн. наук, проф. Кривуля Г.Ф.

Обризан Владимир Игоревич, канд. техн. наук, директор Design & Test Lab. Научные интересы: компьютерная инженерия. Адрес: Украина, 61166, Харьков, ул. Бакулина, 14.

Соклакова Татьяна Игоревна, инженер кафедры АПБТ ХНУРЭ. Научные интересы: компьютерная инженерия. Адрес: Украина, 61166, Харьков, пр. Науки, 14.

Obryzan Vladimir Igorevich, PhD, director of Design & Test Lab. Scientific interests: computer engineering. Address: Ukraine, 61166, Kharkov, ul. Bakulina, 14.

Soklakov Tetiana Igorevna, Engineer, Design Automation Department. Scientific interests: computer engineering. Address: Ukraine, 61166, Kharkov, Nauki Ave, 14.

ІНФОРМАЦІЙНО-ЕКСТРЕМАЛЬНИЙ АЛГОРИТМ НАВЧАННЯ СИСТЕМИ ФУНКЦІОНАЛЬНОГО КОНТРОЛЮ ЕЛЕКТРОПРИВОДУ ШАХТНОЇ ПІДЙОМНОЇ МАШИНИ

ДОВБИШ А.С., ЗИМОВЕЦЬ В.І., КОЗЛОВ З.С.

Розглядається інформаційно-екстремальний алгоритм навчання системи керування електроприводом шахтної підйомної машини. У процесі машинного навчання здійснювалася оптимізація системи контрольних допусків на ознаки розпізнавання за модифікованим інформаційним критерієм Кульбака. Запропонований алгоритм реалізовано на прикладі розв'язання задачі функціонального контролю електроприводу шахтної підйомної машини.

Вступ

У гірничодобувній промисловості для підвищення функціональної ефективності керування технологічними процесами й обладнанням широко впроваджуються інтелектуальні інформаційні технології аналізу даних. Однією з важливих функцій існуючих інтелектуальних автоматизованих систем керування є прогнозування технічного стану гірничого обладнання з метою продовження терміну його служби. Це дозволяє організувати технічне обслуговування обладнання по його дійсному стану, а не за усередненими результатами напрацювання на відмову або загального часу експлуатації [1 – 3]. Необхідною умовою розв'язання цієї важливої науково-технічної задачі є впровадження функціонального контролю для забезпечення надійної експлуатації обладнання шахтної підйомної машини, оскільки його відмови можуть призвести до важких техногенних наслідків.

Найбільш поширеним апаратно-програмним інструментарієм створення інтелектуальних систем керування є штучні нейронні мережі. Але перехід на парадигму прогностичного керування пов'язаний з необхідністю аналізу великих масивів даних, що ускладнює використання штучних нейронних мереж через їх чутливість до багатовимірності алфавіту класів розпізнавання та словника ознак. Тому на практиці підвищення достовірності розпізнавання досягається двома шляхами:

- збільшення глибини машинного навчання штучної нейронної мережі, яка визначається кількістю її шарів;
- редукція простору ознак розпізнавання.

Обидва ці шляхи не гарантують підвищення функціональної ефективності машинного навчання, оскільки пов'язані із втратою інформації.

Одним із перспективних напрямків підвищення функціональної ефективності керування приводом шахтної підйомної машини є використання

ідей і методів так званої інформаційно-екстремальної інтелектуальної технології (ІЕІ – технологія), яка ґрунтується на максимізації інформаційної спроможності системи керування шляхом введення в процесі її навчання додаткових інформаційних обмежень [4, 5]. Основна перевага ІЕІ-технології на відміну від штучних нейронних мереж полягає у використанні геометричного підходу до побудови вирішальних правил, що робить її практично інваріантною до вимірності простору ознак розпізнавання. В роботі [6] розглядалося застосування базового інформаційно-екстремального алгоритму машинного навчання для функціонального контролю електроприводу шахтної підйомної машини. При цьому одержані результати не забезпечили високої функціональної ефективності машинного навчання через апріорно неоптимальну систему контрольних допусків на ознаки розпізнавання.

У статті з метою підвищення функціональної ефективності навчання системи функціонального контролю електроприводу, яка є складовою інтелектуальної автоматизованої системи керування шахтної підйомної машини, пропонується алгоритм інформаційно-екстремального машинного навчання з оптимізацією контрольних допусків на ознаки розпізнавання.

1. Постановка задачі

Нехай сформовано алфавіт класів розпізнавання $\{X_m^0 \mid m=1, M\}$, які характеризують відповідні функціональні стани вузлів електроприводу шахтної підйомної машини, і відповідну вхідну багатовимірну (векторну) навчальну матрицю типу «об'єкт-властивість»:

$$\|y_{m,i}^{(j)} \mid i=1, N; j=1, n\|,$$

де N, n – відповідно кількість ознак розпізнавання у векторі-реалізації (далі просто реалізація) образу і кількість реалізацій, яка дорівнює кількості циклів підйому-спуску. Таким чином, кожний стовпчик матриці $\|y_{m,i}^{(j)}\|$ містить значення навчальної вибірки відповідної ознаки розпізнавання, а j -й рядок є реалізацією з N діагностичними ознаками.

Крім того, задано вектор параметрів машинного навчання

$$g = \langle x_m, d_m, \delta \rangle, \quad (1)$$

де x_m – усереднена реалізація класу $\{X_m^0\}$; d_m – радіус контейнера класу $\{X_m^0\}$, що в процесі машинного навчання відновлюється в радіальному базисі простору ознак розпізнавання; δ – параметр симетричного поля контрольних допу-

сків на ознаки розпізнавання, який дорівнює його половині.

На параметри машинного навчання задано такі обмеження: x_m – вектор, вершина якого визначає геометричний центр контейнера класу $\{X_m^0\}$;

$$d_m \in [0; d(x_m \oplus x_c) - 1],$$

де $d(x_m \oplus x_c)$ – кодова відстань центру класу $\{X_m^0\}$ від центру найближчого (сусіднього) до нього класу X_c^0 ;

$$\delta \in [0; \delta_n / 2],$$

тут δ_n – нормоване (експлуатаційне) поле допусків, яке визначає область значень параметра δ .

На етапі машинного навчання необхідно:

1) оптимізувати параметри вектора (1) за усередненим інформаційним критерієм

$$\bar{E} = \frac{1}{M} \sum_{m=1}^M \max_{G_E \cap \{k\}} E_m^{(k)}, \quad (2)$$

де $E_m^{(k)}$ – інформаційний критерій оптимізації параметрів машинного навчання, який обчислюється на k -му кроці навчання; G_E – робоча (допустима) область визначення функції інформаційного критерію; $\{k\}$ – множина кроків машинного навчання;

2) за оптимальними геометричними параметрами побудувати вирішальні правила, які гарантують при функціонуванні системи діагностування безпосередньо в робочому режимі високу повну ймовірність прийняття правильних класифікаційних рішень.

2. Опис алгоритму машинного навчання

Оптимізація системи контрольних допусків на ознаки розпізнавання в методах ІЕІ-технології відіграє важливу роль, оскільки вони безпосередньо впливають на геометричні параметри контейнерів класів розпізнавання і, отже, на точнісні характеристики класифікаційних рішень. Крім того, специфіка методів інформаційно-екстремального машинного навчання полягає в перетворенні вхідної евклідової навчальної матриці в робочу бінарну, при якому контрольні допуски на ознаки розпізнавання розглядаються як рівні квантування.

На рис. 1 показано симетричне (двобічне) поле допусків на значення i -ї ознаки y_i , $i = \overline{1, N}$.

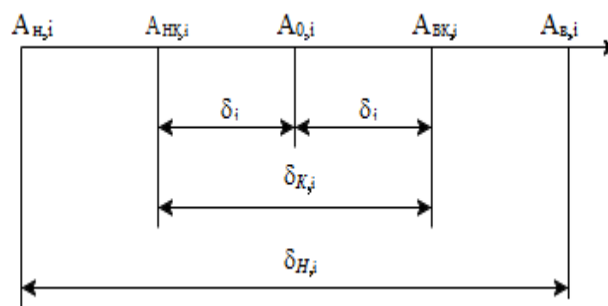


Рис. 1. Симетричне поле допусків

На рис. 1 прийнято такі позначення: $A_{0,i}$ – номінальне значення ознаки; $A_{H,i}$, $A_{B,i}$ – нижній і верхній нормовані допуски відповідно; $A_{HK,i}$, $A_{BK,i}$ – нижній і верхній контрольні допуски відповідно; $\delta_{H,i}$ – нормоване поле допусків; $\delta_{K,i}$ – контрольне поле допусків.

Розглянемо інформаційно-екстремальний алгоритм навчання системи функціонального контролю вузлів електроприводу шахтної підйомної машини з паралельною оптимізацією контрольних допусків на ознаки розпізнавання.

Як відомо, оптимізація контрольних допусків у процесі інформаційно-екстремального машинного навчання здійснюється шляхом пошуку на кожному кроці навчання глобального максимуму інформаційного критерію за двоцикличною ітераційною процедурою:

$$\delta_{K,i}^* = \arg \max_{G_{\delta,i}} \{ \max_{G_E \cap \{k\}} \bar{E}^{(k)} \}, \quad (3)$$

де $G_{\delta,i}$ – допустима область значень параметра δ_i поля контрольних допусків i -ї ознаки розпізнавання.

Процедуру (3) реалізуємо за схемою паралельної оптимізації контрольних допусків, основною перевагою якої є висока оперативність алгоритму машинного навчання у порівнянні з послідовною оптимізацією. Це пов'язано з тим, що контрольні допуски на кожному кроці навчання змінюються одночасно для всіх ознак розпізнавання.

Вхідними даними для алгоритму паралельної оптимізації є масив реалізацій навчальної матриці $\{y_{m,i}^{(j)} | m = \overline{1, M}, i = \overline{1, N}, j = \overline{1, n}\}$, сформованої для заданого алфавіту класів розпізнавання $\{X_m^0\}$, і нормовані допуски $\{\delta_{H,i}\}$, які визначають області значень контрольних допусків відповідних ознак розпізнавання. При цьому задано базовий клас розпізнавання X_1^0 , відносно якого обчислюються контрольні допуски на ознаки розпізнавання. На практиці в задачах функціонального контролю за базовий приймається клас розпізнавання, який характеризує нормальний режим технологічного процесу.

Розглянемо основні етапи реалізації алгоритму навчання СППР з паралельною оптимізацією контрольних допусків на ознаки розпізнавання:

- 1) ініціалізація лічильника класів розпізнавання;
- 2) ініціалізація лічильника кроків зміни параметра δ поля контрольних допусків;
- 3) формування бінарної навчальної матриці $\|x_{m,i}^{(j)}\|$, елементи якої визначаються за правилом

$$x_{m,i}^{(j)} = \begin{cases} 1, & \text{if } y_{m,i}^{(j)} \in \delta_{K,i}; \\ 0, & \text{if } y_{m,i}^{(j)} \notin \delta_{K,i}; \end{cases}$$

- 4) обчислення усередненої двійкової реалізації $\{x_{m,i} | i = \overline{1, N}\}$ класів розпізнавання за правилом

$$x_{m,i} = \begin{cases} 1, & \text{if } \frac{1}{n} \sum_{j=1}^n x_{m,i}^{(j)} > \rho_m; \\ 0, & \text{if } \text{else}, \end{cases}$$

де ρ_m – рівень селекції координат двійкової усередненої реалізації $x_m \in X_m^0$, який за замовчуванням дорівнює 0,5;

- 5) якщо $m \leq M$, то виконується пункт 6, інакше – пункт 1;
- 6) розбиття множини $\{x_{m,i}\}$ на пари усереднених реалізацій сусідніх класів розпізнавання

$$\mathcal{R}_m^{[2]} = \langle x_m, x_c \rangle, \quad (4)$$

де x_c – усереднена реалізація сусіднього класу X_c^0 ;

- 7) обнуління лічильника кроків навчання: $k := 0$;
- 8) ініціалізація лічильника кроків навчання: $k := k + 1$;
- 9) оптимізація радіуса гіперсферичного контейнера класу розпізнавання – кодової відстані d_m відбувається за навчальними матрицями класів розпізнавання, які визначаються відповідними елементами розбиття (4), за ітераційною процедурою пошуку глобального максимуму інформаційного КФЕ (2) в робочій області визначення його функції:

$$d_m^* = \arg \max_{G_E \cap \{k\}} \bar{E}^{(k)}; \quad (5)$$

- 10) якщо $k < d(x_m \oplus x_c)$, то виконується пункт 8, інакше – пункт 11;

- 11) знаходиться в робочій (допустимій) області G_E максимальне значення критерію (2)

$$\bar{E}^* = \max_{G_E \cap \{k\}} \bar{E}^{(k)};$$

- 12) визначаються екстремальний крок навчання k^* , параметр поля контрольних допусків $\delta^* := k^*$ та обчислюються для всіх ознак розпізнавання оптимальні нижні і верхні контрольні допуски на ознаки розпізнавання:

$$A_{HK_i}^* = y - \delta^*; \quad A_{BK_i}^* = y_i + \delta^*;$$

13) ЗУПИН.

Як інформаційний критерій оптимізації параметрів навчання в методах ІЕІ-технології може використовуватися будь-яка інформаційна міра. Але найбільш поширеними є модифіковані інформаційні міри Кульбака та Шеннона. Далі будемо розглядати модифікацію інформаційної міри Кульбака, яка має вигляд [5]

$$E_{Km}^{(k)} = \log_2 \left(\frac{2 - (\alpha_m^{(k)}(d) + \beta_m^{(k)}(d))}{\alpha_m^{(k)}(d) + \beta_m^{(k)}(d)} \right) * [1 - (\alpha_m^{(k)}(d) + \beta_m^{(k)}(d))] \quad (6)$$

де $\alpha_m^{(k)}(d)$ – помилка першого роду прийняття рішення на k -му кроці навчання; $\beta_m^{(k)}(d)$ – помилка другого роду; $D_{1,m}^{(k)}(d)$ – перша достовірність; $D_{2,m}^{(k)}(d)$ – друга достовірність; d – дистанційна міра, яка визначає радіуси гіперсферичних контейнерів, побудованих в радіальному базисі простору Хеммінга.

У загальному випадку залежно від потужності алфавіту класів розпізнавання може бути декілька робочих областей визначення функції інформаційного критерію. При цьому екстремальний крок навчання визначається в робочій області, де глобальне значення інформаційного критерію оптимізації є максимальним.

Таким чином, параметрами інформаційно-екстремального навчання СППР є оптимальні радіуси $\{d_m^*\}$ контейнерів класів розпізнавання, оптимальні еталонні вектори-реалізації $\{x_m^*\}$ класів розпізнавання і оптимальний параметр δ^* поля контрольних допусків, які дозволяють побудувати в просторі ознак розпізнавання вирішальні правила для прийняття рішень при функціонуванні системи в робочому режимі.

3. Приклад реалізації алгоритму машинного навчання

Реалізація запропонованого алгоритму здійснювалася на прикладі машинного навчання системи функціонального контролю електроприводу, яка є складовою автоматизованої системи керування шахтною підйомною машиною.

Структурну схему шахтної підйомної машини показано на рис. 2 [3].

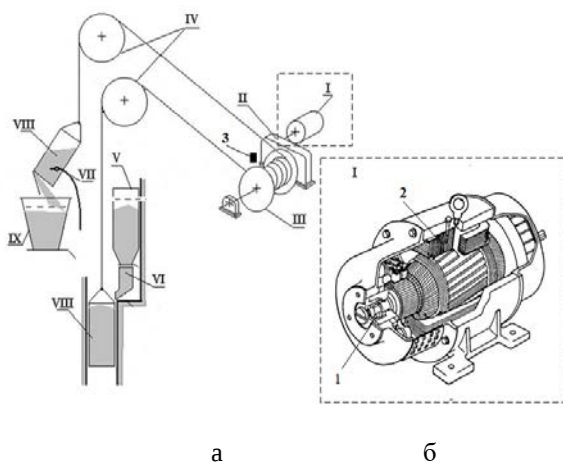


Рис. 2. Структурна схема шахтної підйомної машини: а – загальна схема: I - двигун підйому; II - редуктор; III - барабан; IV - копрові блоки; V - підземний бункер для корисних копалин; VI - ваговий дозатор підземного бункера; VII - розвантажувальні криві; VIII - процес розвантаження скіпа; IX - поверхневий бункер для корисної копалини; б – електропривід: 1 – датчики температури підшипника двигуна; 2 – датчики температури підшипника двигуна; 3 – датчик температури підшипника барабана

Формування навчальної матриці здійснювалося на технологічному циклі спуск-підйом функціонування ШПМ шляхом зчитування інформації з 30 датчиків з інтервалом 200 ms. В результаті кожна реалізація складалася із 60 структурованих ознак розпізнавання. Алфавіт складався з трьох класів розпізнавання: клас X_1^0 характеризував нормальний режим функціонування електроприводу; клас X_2^u – стан підвищеної температури підшипників; X_3^o – стан підвищеної температури обмоток двигуна. Оскільки ознаки розпізнавання мали різні шкали виміру, було виконано їх нормалізацію за методом зведених шкал.

З метою оцінки впливу системи контрольних допусків на ознаки розпізнавання на функціональну ефективність машинного навчання спочатку було реалізовано базовий алгоритм. На рис. 3 показано графіки залежності інформаційного критерію (6) від радіусів контейнерів класів розпізнавання, отримані при реалізації базового алгоритму машинного навчання при неоптимальному параметрі $\delta = \pm 25$.

Аналіз рис. 3 показує, що отримані значення ненормованого критерію (6) є недостатньо високими, що обумовлено високим ступенем перетину класів розпізнавання. При цьому середнє значення критерію оптимізації дорівнює $\bar{E}^* = 1,17$.

Для підвищення функціональної ефективності машинного навчання системи функціонального контролю було застосовано алгоритм паралельної оптимізації контрольних допусків на ознаки розпізнавання, при якому на кожному кроці нав-

чання допуски для всіх ознак розпізнавання змінювалися одночасно.

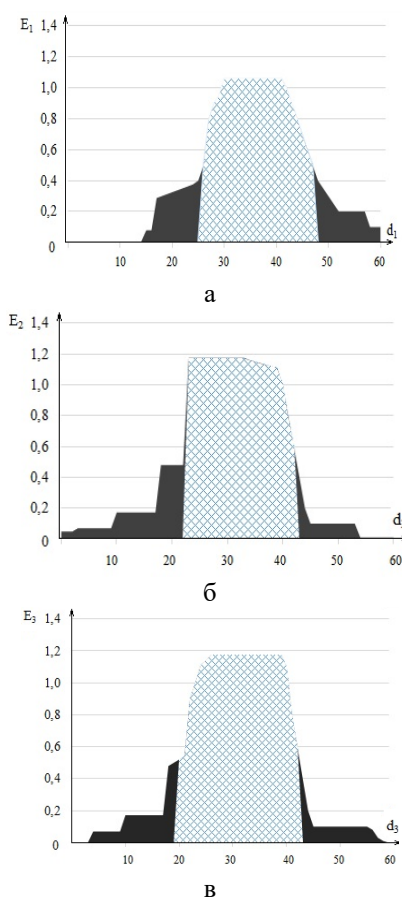


Рис. 3. Графіки зміни критерію (6) при оптимізації радіусів контейнерів класів розпізнавання: а – клас X_1^0 ; б – клас X_2^0 ; в – клас X_3^0

На рис. 4 показано графік залежності усередненого за алфавітом класів розпізнавання КФЕ від параметра δ поля контрольних допусків на ознаки розпізнавання, одержаний у процесі оптимізації системи діагностування вузлів електроприводу ШПМ за алгоритмом паралельної оптимізації.

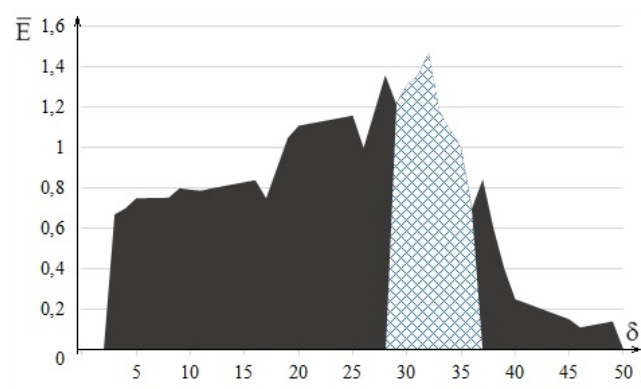


Рис. 4. Графік залежності інформаційного критерію оптимізації від параметра поля контрольних допусків

Аналіз рис. 4 показує, що оптимальне значення параметра поля контрольних допусків дорівнює $\delta^* = \pm 33$ відносних одиниць при максимальному значенні усередненого за алфавітом класів розпізнавання інформаційного критерію (6) $\bar{E}_{\max} = 1,44$, яке обчислювалося в робочій області визначення його функції. Одержане максимальне значення інформаційного критерію перевершує його значення, отримане при реалізації базового інформаційно-екстремального алгоритму машинного навчання. Цей факт свідчить про суттєвий вплив системи контрольних допусків на ознаки розпізнавання на функціональну ефективність машинного навчання.

Для побудови вирішальних правил необхідно було визначити оптимальні геометричні параметри контейнерів класів розпізнавання. На рис. 5 показано графіки залежності критерію оптимізації (6) від радіусів контейнерів класів розпізнавання, одержані при оптимальному значенні параметра поля контрольних допусків на ознаки розпізнавання.

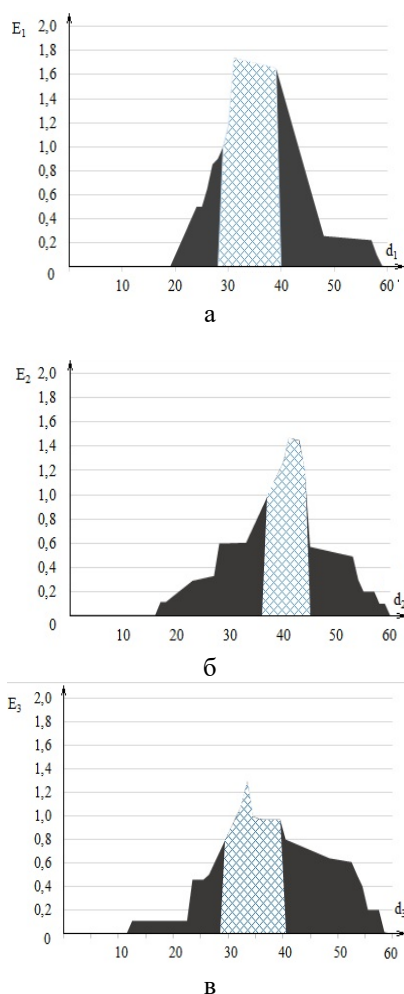


Рис. 5. Графіки залежності інформаційного критерію від радіусів контейнерів класів розпізнавання:

а – клас X_1^0 ; б – клас X_2^0 ; в – клас X_3^0

Аналіз рис. 5 показує, що оптимальні значення радіусів контейнерів класів розпізнавання дорівнюють відповідно: для класу X_1^0 – $d_1^* = 28$ (тут і далі в кодових одиницях); для класу X_2^0 – $d_2^* = 41$ і для класу X_3^0 – $d_3^* = 33$.

За отриманими в процесі інформаційно-екстремального машинного навчання оптимальними геометричними параметрами контейнерів класів розпізнавання було побудовано вирішальні правила, за якими знакододатня функція належності відносить реалізацію, що розпізнається, до того класу розпізнавання із заданого алфавіту, для якого вона максимальна. При цьому функція належності реалізації до гіперсферичного контейнера, наприклад, класу X_m^0 має такий простий вигляд:

$$\mu_m = 1 - \frac{d(x_m^* \oplus x_e)}{d_m^*},$$

де $d(x_m \oplus x_e)$ – кодова відстань між усередненим вектором x_m^* , вершина якого визначає геометричний центр оптимального контейнера класу X_m^0 , і реалізацією x_e , що розпізнається.

Таким чином, оптимізація системи контрольних допусків на ознаки розпізнавання дозволила підвищити функціональну ефективність машинного навчання системи функціонального контролю електроприводу ШПИ у порівнянні з результатами реалізації базового інформаційно-екстремального алгоритму при неоптимальних контрольних допусках. Але в процесі машинного навчання не вдалося побудувати безпомилкові за навчальною матрицею вирішальні правила через суттєвий перетин класів розпізнавання в просторі ознак розпізнавання. Тому для підвищення функціональної ефективності доцільно збільшити глибину машинного навчання шляхом оптимізації інших параметрів навчання.

Висновки

Запропонований інформаційно-екстремальний метод машинного навчання системи функціонального контролю електроприводу шахтної підйомної машини з паралельною оптимізацією контрольних допусків на ознаки розпізнавання дозволяє будувати високостовірні вирішальні правила. Але через суттєвий перетин класів розпізнавання в просторі ознак, що має місце в практичних задачах функціонального контролю, не вдалося побудувати безпомилкові за навчальною матрицею вирішальні правила. Тому в подальшому необхідно дослідити вплив на функціональну ефективність машинного навчання інших параметрів і розробити інформаційно-екстремальні алгоритми їх оптимізації.

Література: 1. *Кауцич А., Марищенко А.* Система управления и комплексной защиты шахтной подъёмной установки / Журнал «Современные технологии автоматизации», 2005. № 2. С. 26 - 28. 2. *Трубецкой К. Н.* Современные системы управления горно-транспортными комплексами / К.Н. Трубецкой, А.А. Кулешов, А.Ф. Клебанов, Д.Я. Владимиров // Под ред. акад. К.Н. Трубецкого. СПб.: Наука, 2007. 306 с. 4. *Довбиш А.С.* Основы проектирования интеллектуальных систем / А.С. Довбиш. Суми: СумДУ, 2009. 171 с. 5. *Довбиш А.С.* Интеллектуальные технологии в электронному навчанні / А. С. Довбиш, А. В. Васильев, В. О. Любчак. Суми: Сумський державний університет, 2013. 178 с. 6. *Зимовець В. І.* Система функціонального контролю приводу шахтної підйомної машини, яка здатна навчатися / В. І. Зимовець, А. С. Чирва, О. І. Марищенко // Журнал інженерних наук. 2016. № 2. С. 15 – 19.

Transliterated bibliography:

1. *Kashhich A., Marishhenko A.* Sistema upravlenija i kompleksnoj zashhity shahtnoj pod#jomnoj ustanovki / Zhurnal «Sovremennye tehnologii avtomatizacii», 2005. № 2. S. 26 - 28.
2. *Trubeckoj K. N.* Sovremennye sistemy upravlenija gorno-transportnymi kompleksami / K.N. Trubeckoj, A.A. Kuleshov, A.F. Klebanov, D.Ja. Vladimirov // Pod red. akad. K.N. Trubeckogo. SPb.: Nauka, 2007. 306 s.
4. *Dovbish A.S.* Osnovi proektuvannja intelektual'nih sistem / A.S. Dovbish. Sumi: SumDU, 2009. 171 c.
5. *Dovbish A.S.* Intelektual'ni tehnologii i elektronnomu navchanni / A. S. Dovbish, A. V. Vasil'ev, V. O. Ljubchak. Sumi: Sums'kij derzhavnij universitet, 2013. 178 s.
6. *Zimovec' V. I.* Sistema funkcional'nogo kontrolju privodu shahtnoï pidjomnoï mashini, zdatna navchatisja / V. I. Zimovec', A. S. Chirva, O. I. Marishhenko // Zhurnal inzhenernih nauk, 2016. № 2. S. 15 – 19.

Надійшла до редколегії 11.11.2017

Рецензент: д-р техн. наук, проф. Кривуля Г.Ф.

Довбиш Анатолій Степанович, д-р техн. наук, професор; завідувач кафедри комп'ютерних наук Сумського державного університету. Наукові інтереси: інтелектуальний аналіз даних. Адреса: Україна, 40007, Суми, вул. Римського-Корсакова, 2.

Зимовець Вікторія Ігорівна, аспірантка кафедри комп'ютерних наук СумДУ. Наукові інтереси: штучний інтелект. Адреса: Україна, 40007, Суми, вул. Римського-Корсакова, 2, e-mail: zc.vika@gmail.com.

Козлов Захар Сергійович, аспірант кафедри комп'ютерних наук СумДУ. Наукові інтереси: машинне навчання та розпізнавання образів. Адреса: Україна, 40007, Суми, вул. Римського-Корсакова, 2.

Dovbish Anatoly Stepanovich, Dr. Tech. Sciences, professor; Head of Computer Science Department, Sumy State University. Scientific interests: intellectual analysis of data. Address: Ukraine, 40007, Sumy, Rimsky-Korsakov st., 2.

Zimovets Victoria I., PhD student, Computer Science Department, Sumy State University.. Scientific interests: artificial intelligence. Address: Ukraine, 40007, Sumy, Rimsky-Korsakov st., 2, e-mail: zc.vika@gmail.com.

Kozlov Zakhar Sergeevich, PhD student, Computer Science Department, Sumy State University. Scientific interests: machin learning and pattern recognition. Address: Ukraine, 40007, Sumy, Rimsky-Korsakov st., 2.

ПРИХОВАНІ МОЖЛИВОСТІ МАТЕМАТИКИ ПРИ ОПРАЦЮВАННІ НЕЧІТКОЇ ІНФОРМАЦІЇ

ЗАЯЦЬ В.М., РИБИЦЬКА О.М., ЗАЯЦЬ М.М.

Для розв'язання прикладних задач обробки статистичної інформації пропонується застосовувати теорію нечітких інтегралів та розмитої логіки. Розроблені підходи забезпечують вдалий вибір функції належності, критерію правдоподібності та нечіткої міри на основі знань та інтуїції експерта. Обґрунтовується доцільність використання декларативних мов програмування для обробки статистичної інформації. Застосування цих підходів ілюструється конкретними прикладами. Вказуються напрями подальшого використання отриманих результатів.

Ключові слова нечітка множина; функція належності, лінгвістична змінна, база нечітких знань, нечітка логіка.
Key words: software, artificial intelligence, declarative programming language, frames, Lisp, Prolog, universal modeling language.

1. Вступ

Проблеми математичного і комп'ютерного моделювання на сьогодні полягають, зокрема, в незастосовності чіткої логіки та моделей задач із чітко визначеними входними параметрами у випадках, коли з якихось причин наявні протиріччя, невизначеність або нечіткість інформації про досліджуваний об'єкт, систему чи явище [1].

Невизначеність, як відомо, виникає через недостатність знань, які стосуються конкретної події [2]. Вона присутня до проведення експерименту. Математична модель невизначеності ґрунтується на теорії ймовірностей, теорії можливостей, мір довіри та ряді інших.

Феномен нечіткості виникає в процесі об'єднання в одне ціле X об'єктів, які мають спільну властивість φ :

$$X = \{x | x \text{ володіє } \varphi\},$$

де x пробігає всі елементи деякої універсальної множини.

Зважаючи на те, що в реальності завжди є елементи x , для яких неясно, чи мають вони вказану властивість, чи ні, X не є множиною в класичному розумінні. Всяка спроба трактувати загальний опис призводить до нечітких понять, оскільки точний опис містить надлишок деталей. Збільшення точності у описі веде до збільшення кількості інформації, змістовність якої зменшується до того моменту, поки точність і змістовність не стають

взаємовиключними. Про необхідність нечіткості для передачі змістовної інформації вперше наголосив Л.А. Заде [3]. Саме ідеї цього американського вченого зробили поштовх для розвитку «нечіткої математики» [4], яка поряд з апаратом нечітких множин містить інші прийоми роботи з невизначеністю.

Застосування теорії нечітких множин та мір – це крок на шляху до зближення точності класичної математики з наповненим неточністю навколишнім середовищем, спроба подолати лінгвістичний бар'єр між людиною, судження і оцінки якої є наближеними та нечіткими, і технічними засобами, які можуть виконувати тільки чіткі інструкції [5].

2. Особливості застосування Fuzzy-технологій до побудови експертних систем

Апаратом, який дозволяє працювати із нечіткою логікою, «розмитими» параметрами моделей, є апарат Fuzzy-технологій. Підрозділом Fuzzy-технологій є нечіткі експертні системи.

Системи нечіткого виводу призначені для реалізації процесу нечіткого виводу і служать концептуальним базисом всієї сучасної нечіткої логіки. Системи нечіткого виводу дозволяють вирішувати завдання автоматичного управління, класифікації даних, розпізнавання образів, ухвалення рішень і багато інших.

Лінгвістичні змінні розширюють можливості представлення знань. Їх визначають через нечіткі множини, значення яких встановлюються функціями належності. Функції належності можна одержати через оцінки експертів [6] або шляхом аналізу нечітких кластерів. Згідно з [6] нечіткі експертні системи можуть бути втілені в життя, коли вартість отримання точної інформації, тобто інформації в абсолютному вимірі, перевищує максимальний дохід від реструктуризації моделі або є практично неможливою.

Відомо, що первісний етап побудови штучного інтелекту на засадах використання природної мови базується на двозначній логіці та механізмі виведення із жорсткими правилами. Експертні системи першого покоління працюють за таким набором правил:

Якщо X належить A_k , тоді Y належить C_k , де $k = \overline{1, N}$.

Тут A_k і C_k є набором точних числових або символічних значень, які можуть бути лінгвістичними значеннями лінгвістичних змінних.

На сучасному етапі в експертних системах для представлення знань найбільш вживаним формалізмом є продукційні моделі

«Якщо A , тоді B ».

Друге покоління експертних систем має принаймні дві особливості: нечітке подання знань та нечітке виведення. Одну з найтипівіших задач логічного виведення за умов нечіткості можна сформулювати так

Дано (нечітке) продукційне правило "Якщо A , то B ".

Спостерігається A' (A в певній мірі). Яким повинно бути B ?

Після отримання нечіткої множини висновків знаходять конкретний числовий відповідник (проводять дефазифікацію).

3. Побудови моделей з розмитотою логікою

Розглянемо об'єкт з одним виходом і n входами типу

$$y = f(x_1, x_2, \dots, x_n), \quad (1)$$

де $x_1, \dots, x_n$ – набір значень вхідних змінних; y – вихідна змінна.

Для побудови математичної моделі на засадах встановлення взаємозв'язку між вхідними і вихідними змінними згідно з експериментальними даними, шляхом проведення операції фазифікації, кількісні і якісні змінні переводять у лінгвістичні терми:

$$U_i = [\underline{u}_i, \overline{u}_i], \quad i = \overline{1, n}, \quad (2)$$

$$Y = [\underline{y}, \overline{y}], \quad (3)$$

де $\underline{u}_i, \overline{u}_i$ – найменше та найбільше можливе значення змінної величини x_i ; $\underline{y}, \overline{y}$ – найменше та найбільше можливе значення вихідної змінної y .

Для розв'язання поставленої задачі (1) необхідно застосувати методику прийняття рішення, за допомогою якої фіксованому вектору вхідних змінних $x^* = (x_1^*, x_2^*, \dots, x_n^*)$, $x_i^* \in U_i$ однозначно ставився б у відповідність розв'язок $y^* \in Y$. Для формального встановлення такої залежності будемо розглядати вхідні x_i , $i = \overline{1, n}$ та вихідний y параметри як лінгвістичні змінні, задані на універсальних множинах (2), (3). Для оцінки лінгвістичних змін-

них x_i , $i = \overline{1, n}$ та y будемо використовувати якісні терми з таких терм-множин:

$A_i = \{a_i^1, a_i^2, \dots, a_i^{p_i}\}$ – терм-множина вхідної змінної

x_i , $i = \overline{1, n}$; $D = \{d_1, d_2, \dots, d_m\}$ – терм-множина вихідної змінної y . Для побудови терм – множин можна застосувати, наприклад, методику, запропоновану в [7].

Для кожного терма кожної лінгвістичної змінної на основі знань експерта будують функції належності $\mu^{a_i^p}(x)$ та $\mu^{d_j}(y)$ (трапецієподібні, трикутні

тощо [5]), де $\mu^{a_i^p}(x)$ – міра належності елемента $x \in U_i$ до терму $a_i^p \in A_i$, $i = \overline{1, n}$; $p = \overline{1, p_i}$; $\mu^{d_j}(y)$ – міра належності елемента $y \in Y$ до терму $d_j \in D$, $j = \overline{1, m}$.

Визначення лінгвістичних оцінок змінних і необхідних для їх формалізації функцій належності є першим етапом побудови нечіткої моделі досліджуваного об'єкта. У літературі з нечіткої логіки він одержав назву *фазифікації змінних* (англ. *fuzzification*) [8].

Наступним кроком є створення бази нечітких знань.

4. Підхід до побудови моделей баз нечітких знань

Нехай для об'єкта (1) відомо N правил, які пов'язують його входи й вихід за допомогою векторів типу:

$$V_k = (x_1, x_2, \dots, x_n, y), \quad k = \overline{1, N},$$

$$\text{причому} \quad N = k_1 + \dots + k_j + \dots + k_m, \quad (4)$$

де k_j – кількість експериментальних даних, що відповідають однаковому значенню d_j терм-множини вихідної змінної y ; m – загальна кількість термів вихідної змінної, причому в загальному випадку $k_1 \neq \dots \neq k_m$.

Згідно з принципом моделювання, вказаним в [9], вважаємо, що $N < p_1 p_2 \dots p_n$, тобто кількість наявних експериментальних даних менша повного перебору різних комбінацій можливих термів вхідних змінних p_i , $i = \overline{1, n}$. Тоді база знань – це таблиця, сформована за такими правилами:

1. Розмірність таблиці дорівнює $N \times (n + 2)$, у якій $n + 2$ – кількість стовпців, а N – кількість рядків.
2. Кожен рядок матриці – це комбінація вхідних змінних, віднесених експертом до одного з можливих значень терм-множини вихідної змінної y . При цьому перші k_1 рядків відповідають значенню вихідної змінної $y = d_1$, наступні k_2 рядки – значенню $y = d_2$, а останні k_m рядків – значенню $y = d_m$;
3. Перші n стовпців матриці відповідають вхідним змінним $x_i, i = \overline{1, n}$; $(n + 1)$ -й – вага правила W_{jp} , $j = \overline{1, m}$, $p = \overline{1, k_j}$, а $(n + 2)$ -й – значенням d_j вихідної терм-множини змінної y , $j = \overline{1, m}$, які відповідають комбінації значень у перших $(n + 1)$ стовпцях.
4. Елемент a_i^{jp} , що знаходиться на перетині i -го стовпця та jp -го рядка, відповідає лінгвістичній оцінці параметра x_i в рядку матриці знань з номером jp . При цьому лінгвістичну оцінку a_i^{jp} вибирають із терм-множини, що відповідає змінній x_i , тобто $a_i^{jp} \in A_i, i = \overline{1, n}; j = \overline{1, m}; p = \overline{1, k_j}$.

При формуванні експертом лінгвістичних правил типу «ЯКЩО – ТОДІ», які утворюють базу нечітких знань про певний об'єкт, впевненість експерта в кожному правилі може бути різною. Якщо одне правило на думку експерта може слугувати як безперечна істина, то стосовно іншого правила в того ж експерта можуть бути деякі сумніви.

З метою відображення цих різних ступенів впевненості в базу нечітких знань вводяться ваги правил – це числа з інтервалу $[0, 1]$, що характеризують впевненість експерта в кожному вибраному ним конкретному правилі для прийняття рішення. Загальний вигляд бази знань наведено в табл. 1.

Після побудови бази знань необхідно ретельно перевірити у табл. 1 наявність протилежних за змістом рядків, тобто правил, що за однакових вхідних змінних мають різні вихідні значення. Введена матриця знань визначає систему логічних висловлювань типу «ЯКЩО – ТОДІ, ІНАКШЕ», які пов'язують значення вхідних змінних $x_1, \dots, x_n$ з одним із можливих значень виходу $d_j, j = \overline{1, m}$:

Таблиця 1. Загальний вигляд бази нечітких знань

Но- мер вхі- дної ком- бі- нації	Вхідні змінні				Вага	Вихі- дна змінна
	x_1	x_2	$\dots x_i \dots$	x_n	w	y
11	a_1^{11}	a_2^{11}	a_i^{11}	a_n^{11}	w_{11}	d_1
12	a_1^{12}	a_2^{12}	a_i^{12}	a_n^{12}	w_{12}	
...	...	...	...	...	...	
$1k_1$	$a_1^{1k_1}$	$a_2^{1k_1}$	$a_i^{1k_1}$	$a_n^{1k_1}$	w_{1k_1}	
...	...	...	...	...	...	...
j_1	a_1^{j1}	a_2^{j1}	a_i^{j1}	a_n^{j1}	w_{j1}	d_j
j_2	a_1^{j2}	a_2^{j2}	a_i^{j2}	a_n^{j2}	w_{j2}	
...	...	...	...	...	...	
jk_j	$a_1^{jk_j}$	$a_2^{jk_j}$	$a_i^{jk_j}$	$a_n^{jk_j}$	w_{jk_j}	
...	...	...	...	...	...	...
m_1	a_1^{m1}	a_2^{m1}	a_i^{m1}	a_n^{m1}	w_{m1}	d_m
m_2	a_1^{m2}	a_2^{m2}	a_i^{m2}	a_n^{m2}	w_{m2}	
...	...	...	...	...	...	
mk_m	$a_1^{mk_m}$	$a_2^{mk_m}$	$a_i^{mk_m}$	$a_n^{mk_m}$	w_{mk_m}	

ЯКЩО $(x_1 = a_1^{11}) \text{ І } (x_2 = a_2^{11}) \text{ І } \dots \text{ І } (x_n = a_n^{11})$ (з вагою w_{11}),

АБО $(x_1 = a_1^{12}) \text{ І } (x_2 = a_2^{12}) \text{ І } \dots \text{ І } (x_n = a_n^{12})$ (з вагою w_{12}),

АБО ...

АБО $(x_1 = a_1^{1k_1}) \text{ І } (x_2 = a_2^{1k_1}) \text{ І } \dots \text{ І } (x_n = a_n^{1k_1})$ (з вагою w_{1k_1}),

ТОДІ $y = d_1$, ІНАКШЕ

ЯКЩО $(x_1 = a_1^{21}) \text{ І } (x_2 = a_2^{21}) \text{ І } \dots \text{ І } (x_n = a_n^{21})$ (з вагою w_{21}),

АБО ...

АБО $(x_1 = a_1^{2k_2}) \text{ І } (x_2 = a_2^{2k_2}) \text{ І } \dots \text{ І } (x_n = a_n^{2k_2})$ (з вагою w_{2k_2}),

ТОДІ $y = d_2$, ІНАКШЕ ...

ЯКЩО $(x_1 = a_1^{m1}) \text{ І } (x_2 = a_2^{m1}) \text{ І } \dots \text{ І } (x_n = a_n^{m1})$ (з вагою w_{m1}),

АБО ...

АБО $(x_1 = a_1^{mk_m}) \text{ I } (x_2 = a_2^{mk_m}) \text{ I } \dots \text{ I } (x_n = a_n^{mk_m})$ (з вагою w_{mk_m}),
ТОДІ $y = d_m$.

Подібну систему логічних висловлювань називають нечіткою базою знань. Із використанням операцій $\cup$ (АБО) й $\cap$ (І) описана система логічних висловлювань може бути переписана в більш компактному вигляді:

$$\bigcup_{p=1}^{kj} \left[\bigcap_{i=1}^n (x_i = a_i^{jp}) \right] \rightarrow y = d_j, \quad j = \overline{1, m}. \quad (5)$$

Таким чином, вхідне співвідношення (1), що встановлює зв'язок між вхідними параметрами x_i , $i = \overline{1, n}$ та вихідною змінною y , формалізовано у вигляді системи нечітких логічних висловлювань (5), що базується на створеній матриці знань. Правила описаної системи нечіткого виведення, у яких ступінь істинності відмінний від нуля, вважають активними.

У роботі [10] було запропоновано метод, що полягає у використанні нечітких логічних рівнянь, які будуються на основі матриці знань або ізоморфної до неї системи логічних висловлювань (5) і дозволяють обчислювати значення функцій належності вихідної змінної за фіксованих значень входів об'єкта.

Лінгвістичні оцінки a_i^{jp} змінних $x_1, \dots, x_n$, що входять у логічні висловлювання (5), будемо розглядати як нечіткі множини, визначені на універсальних множинах (2). Введемо такі позначення: $\mu_{a_i^{jp}}(x_i)$ – функція належності параметра x_i до нечіткого терму a_i^{jp} , $i = \overline{1, n}$, $j = \overline{1, m}$, $p = \overline{1, k_j}$;

$\mu^{dj}(x_1, x_2, \dots, x_n)$ – функція належності вектора вхідних змінних $x = (x_1, x_2, \dots, x_n)$ терму вихідної змінної $y = d_j$, $j = \overline{1, m}$.

Таким чином, у нас наявні два типи функцій, зв'язок між якими визначається базою нечітких знань (5), на основі чого можна вивести систему логічних рівнянь:

$$\mu^{d1}(x_1, x_2, \dots, x_n) = w_{11}[\mu^{a11}(x_1) \wedge \mu^{a21}(x_2) \wedge \dots \wedge \mu^{an1}(x_n)] \vee$$

$$\vee w_{12}[\mu^{a12}(x_1) \wedge \mu^{a22}(x_2) \wedge \dots \wedge \mu^{an2}(x_n)] \vee \dots \vee w_{1k_1}[\mu^{a1k_1}(x_1) \wedge \mu^{a2k_1}(x_2) \wedge \dots \wedge \mu^{an1}(x_n)],$$

$$\mu^{d2}(x_1, x_2, \dots, x_n) = w_{21}[\mu^{a11}(x_1) \wedge \mu^{a21}(x_2) \wedge \dots \wedge \mu^{an1}(x_n)] \vee \dots \vee w_{2k_2}[\mu^{a1k_2}(x_1) \wedge \mu^{a2k_2}(x_2) \wedge \dots \wedge \mu^{an2}(x_n)],$$

(6)

$$\dots \vee w_{m1}[\mu^{a1m1}(x_1) \wedge \mu^{a2m1}(x_2) \wedge \dots \wedge \mu^{anm1}(x_n)] \vee \dots \vee w_{m2}[\mu^{a1m2}(x_1) \wedge \mu^{a2m2}(x_2) \wedge \dots \wedge \mu^{anm2}(x_n)] \vee \dots \vee w_{mk_m}[\mu^{a1mk_m}(x_1) \wedge \mu^{a2mk_m}(x_2) \wedge \dots \wedge \mu^{anmk_m}(x_n)],$$

де $\vee$ – логічне «АБО»; $\wedge$ – логічне «І».

За аналогією з (5), дану систему (6) нечітких логічних рівнянь можна подати в компактнішому вигляді:

$$\mu^{dj}(x_1, x_2, \dots, x_n) = \bigvee_{p=1}^{kj} \left(w_{jp} \left[\bigwedge_{i=1}^n \mu_{a_i^{jp}}(x_i) \right] \right), \quad j = \overline{1, m}. \quad (7)$$

Прийняття рішення $d^* \in D\{d_1, d_2, \dots, d_m\}$, яке відповідає фіксованому вектору значень вхідних змінних $x^* = (x_1^*, x_2^*, \dots, x_n^*)$, будемо здійснювати відповідно до алгоритму [10], побудованого з використанням апарату нечіткої логіки:

1. Визначається можливий діапазон змінювання контрольованих параметрів, складається база знань з використанням експертних даних та виводиться система нечітких логічних рівнянь (7).

2. Фіксується вектор значень вхідних змінних $x^* = (x_1^*, x_2^*, \dots, x_n^*)$.

3. Задається вигляд функцій належності нечітких термів для різних контрольованих параметрів.

4. Використовуючи логічні рівняння (7), обчислюються значення багатопараметричних функцій належності $\mu^{dj}(x_1^*, x_2^*, \dots, x_n^*)$ вектора x для всіх значень d_j , $j = \overline{1, m}$ вихідної змінної y . При цьому логічні операції $\vee$ (АБО) й $\wedge$ (І) над функціями належності замінюються на операції $\max$ та $\min$:

$$\mu(a) \vee \mu(b) = \max[\mu(a), \mu(b)], \quad (8)$$

$$\mu(a) \wedge \mu(b) = \min[\mu(a), \mu(b)]. \quad (9)$$

Отже, спочатку знаходять мінімальні значення функцій належності в кожному правилі, а потім поміж них обирають найбільше значення функції належності серед усіх правил для кожного значення d_j , $j = \overline{1, m}$, яке і ставлять у відповідність вихідній змінній y . Таким чином робиться висновок про належність вихідної змінної y до терму d_j^* , функція належності якого максимальна.

Запропонований алгоритм використовує ідею ідентифікації лінгвістичного терму за максимумом функції належності та узагальнює цей підхід на всю матрицю знань. Обчислювальна частина даного алгоритму легко реалізується шляхом простого застосування операцій *max* та *min*.

Для одержання чіткого числа з інтервалу $[y, \bar{y}]$, що відповідає нечіткому значенню вихідної змінної, необхідно застосувати операцію дефазифікації. Визначити це чітке число y^* можна, наприклад, за методом центру тяжіння:

$$y^* = \frac{\int_{Min}^{Max} y \mu(y) dy}{\int_{Min}^{Max} \mu(y) dy}, \quad (10)$$

де *Min* і *Max* – ліва і права точки інтервалу носія нечіткої множини вихідної змінної y .

5. Задача прогнозування вартості товару

В роботі на основі запропонованого підходу розглянута задача прогнозування вартості нереалізованого товару певного виду на кінець сезону торгівлі торговельної фірми, яка займається препаратами хімічного захисту рослин.

Експертом було встановлено, що вагомими чинниками, які впливають на залишок Y , є: x_1 («залишок») – залишок попереднього сезону торгівлі (у доларах США); x_2 («нові закупки») – вартість нових закупок (у доларах США); x_3 («націнка») – середня величина торговельної націнки (у відсотках); x_4 («термін») – тривалість продажу даного препарату (у роках). Універсальні множини для описаних змінних визначено такими: $U_1 = [0; 600000]$; $U_2 = [20000; 1500000]$; $U_3 = [0; 50]$; $U_4 = [0; 10]$. Універсальна множина для

прогнозованої величини збігається, очевидно, з U_1 .

Для кожної вхідної та вихідної змінних побудовано терм-множини:

$$A_1 = \{ \text{"малий"}, \text{"середній"}, \text{"великий"}, \text{"критичний"} \} = \{ M, C, B, K \};$$

$$A_2 = \{ \text{"мала"}, \text{"середня"}, \text{"велика"} \} = \{ M, C, B \};$$

$$A_3 = \{ \text{"малий"}, \text{"середній"}, \text{"великий"} \} = \{ M, C, B \};$$

$$A_4 = \{ \text{"короткий"}, \text{"середній"}, \text{"довготривалий"} \} = \{ K, C, D \};$$

$$D = \{ \text{"малий"}, \text{"середній"}, \text{"великий"}, \text{"критичний"} \} = \{ M, C, B, K \}.$$

Виходячи з інформації, наданої експертом, побудовано функції належності для термів вхідних та вихідної змінних (рис.1-4).

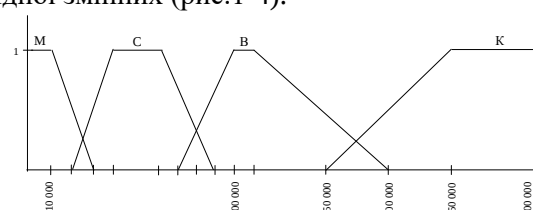


Рис.1 Функція належності лінгвістичної змінної «залишок»

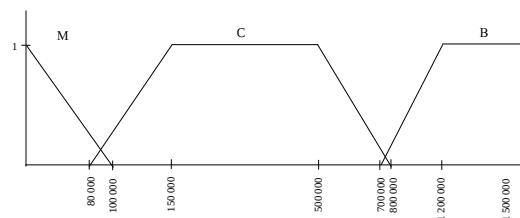


Рис.2. Функція належності лінгвістичної змінної «нові закупки»

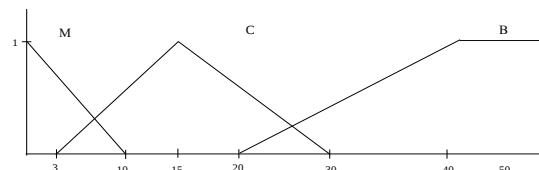


Рис. 3. Функція належності лінгвістичної змінної «націнка»

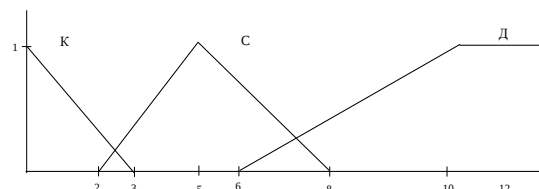


Рис.4. Функція належності лінгвістичної змінної «термін»

Наступним кроком є побудова бази нечітких знань (табл.2).

Таблиця 2. База нечітких знань задачі

Номер вхід- ної ком- біна- ції (логі- чного пра- вила)	Вхідні змінні				Ва- га	Ви- хід- на змін- на
	x_1	x_2	x_3	x_4	w	y
11	М	М	М	К	1	d_1
12	В	М	М	С	0,9	
13	В	М	М	Д	1	
14	К	М	М	С	1	
15	М	М	В	Д	0,7	
16	М	С	В	С	0,7	
17	С	М	С	С	0,8	
18	С	М	С	Д	0,8	
19	С	С	С	С	0,7	
20	В	М	С	Д	0,9	
21	В	М	В	Д	0,5	
23	С	М	В	Д	1	d_2
24	М	С	С	С	1	
25	М	В	В	С	0,8	
26	С	С	М	С	0,8	
27	С	В	М	В	0,7	
28	С	В	С	С	0,7	
29	В	С	С	С	0,7	d_3
30	В	С	С	Д	0,5	
31	В	С	В	Д	0,3	
32	К	С	В	Д	0,8	
33	М	В	С	С	0,9	
34	М	В	С	К	1	
35	С	В	М	К	1	
36	С	В	С	Д	0,8	
37	М	В	В	С	0,8	d_4
38	М	В	В	Д	0,9	
39	С	В	В	С	0,9	
40	С	В	В	Д	1	
41	В	В	В	Д	1	

Наступні розрахунки проведено для даних торго-
вельного підприємства за сезон продажів певного
року: $x_1^* = 80000$; $x_2^* = 36000$; $x_3^* = 22$; $x_4^* = 9$. У
цьому випадку змінна x_1 належить до термів «се-
редній» (з мірою належності
 $\mu(x) = 1 - \frac{1}{30000}(x - 60000)$) або «високий» (із мірою
належності $\mu(x) = \frac{1}{30000}(x - 70000)$); x_2 – терму

«мала» (з мірою належності $\mu(x) = 1 - \frac{1}{100000}x$);

x_3 – до термів «середня» (з мірою належності

$\mu(x) = 1 - \frac{1}{15}(x - 15)$) або «висока» (з мірою нале-

жності $\mu(x) = \frac{1}{20}(x - 20)$); x_4 – до терма «дов-

готривалий» (із мірою належності

$\mu(x) = \frac{1}{4}(x - 6)$). Простим співставленням легко

побачити, що активними будуть правила 18, 20, 21
(приводять до виходу d_1) та 23 (приводить до ви-
ходу d_2).

Із відповідної системи логічних рівнянь одержимо
значення μ^{d_1} та μ^{d_2} :

$$\mu^{d_1} = \max \{0,8 \min \{0,333; 0,64; 0,53; 0,75\}; 0,9 \min \{0,333; 0,64; 0,53; 0,75\}; 0,5 \min \{0,333; 0,64; 0,1; 0,75\}\} = \max \{0,2664; 0,3; 0,05\} = 0,3;$$

$$\mu^{d_2} = \max \{\min \{0,333; 0,64; 0,1; 0,75\}\} = 0,1.$$

Після проведення описаних в алгоритмі обчислень
одержимо міру належності вихідної змінної:

$$\mu(w) = \begin{cases} 0,3, & \text{якщо } w \in (0; 24000); \\ 1 - \frac{1}{20000}(w - 10000), & \text{якщо } w \in (24000; 28000); \\ 0,1, & \text{якщо } w \in (28000; 87000); \\ 1 - \frac{1}{30000}(w - 60000), & \text{якщо } w \in (87000; 90000). \end{cases}$$

Графік отриманої функції належності зображено
на рис.5.

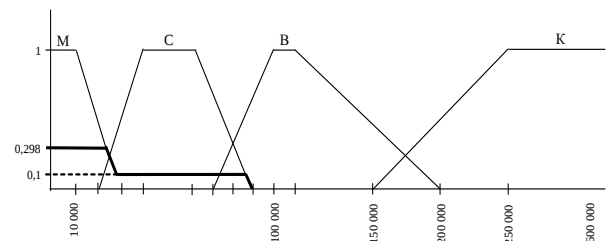


Рис. 5. Графік функції належності вихідної змінної «за-
лишок»

Кількісне значення вихідної величини y^* (резуль-
тат дефазифікації) обчислимо методом центру
тяжіння (10):

$$y^* = \frac{\int_{Min}^{Max} w\mu(w)dw}{\int_{Min}^{Max} \mu(w)dw}.$$

$$\begin{aligned} \int_{Min}^{Max} w\mu(w)dw &= \int_0^{24000} 0,3wdw + \int_{24000}^{28000} w\left(1 - \frac{1}{20000}(w-10000)\right)dw + \\ &+ \int_{28000}^{87000} 0,1dw + \int_{87000}^{90000} w\left(1 - \frac{1}{30000}(w-60000)\right)dw = 172800000 + \\ &+ 1398400 + 339250000 + 131999500 = 353865180; \\ \int_{Min}^{Max} \mu(w)dw &= \int_0^{24000} 0,3dw + \int_{24000}^{28000} \left(1 - \frac{1}{20000}(w-10000)\right)dw + \\ &+ \int_{28000}^{87000} 0,1dw + \int_{87000}^{90000} \left(1 - \frac{1}{30000}(w-60000)\right)dw = 7200 + \\ &+ 800 + 5500 + 88,5 = 13588,5. \end{aligned}$$

Остаточного значення прогнозованої величини залишку товарної маси дорівнює

$$y^* = \frac{353865180}{13588,5}; 26041,5$$

що є достатньо близьким до реального залишку товару на кінець сезону вибраного року, а саме 23200 доларів США.

Слід зауважити, що прогнозовані значення можна пробувати знаходити більш близькими до реально спостережуваних (за декілька попередніх років) шляхом перегляду встановлених вагових коефіцієнтів, коригування функцій належності тощо. Також можна збільшувати кількість входних величин. Проте при їх надто великій кількості побудова нечіткої бази знань про невідому залежність стає важким завданням. Це обумовлене тим, що в пам'яті середньої статистичної людини одночасно може утримуватись не більше за 7 ± 2 понять-ознак. У таких випадках доцільно проводити класифікацію входних змінних і згідно з нею будувати дерево виведення, яке визначає систему вкладених один в одного висловлювань [5].

Можна запропонувати інший підхід до опрацювання великих обсягів нечітких даних за умов неповної визначеності вектора входних змінних (первинних ознак). Суть підходу ґрунтується на проведенні імітаційного моделювання поведінки досліджуваної системи і шляхом експертної оцінки доповнення існуючої бази знань новими інформативними даними та встановлення вектора входних ознак. Очевидно, такий підхід є ітераційним і необхідно потурбуватися про збіжність процесу вирахувань для досягнення поставленої мети при мінімальних затратах та обмеженій похибці.

6. Висновки

Для роботи з нечисловими даними в задачах штучного інтелекту, побудови систем розпізнавання,

експертних систем, медичною та параметричною діагностикою, створення логіко-лінгвістичних моделей найбільш вдало пристосовані декларативні мови програмування, які на мові логічних висловлювань та функціонально-логічних залежностей забезпечують можливість опису задачі з нечітко формульованими даними і отримання розв'язків у вигляді логічних вислідів, нових функціональних залежностей або імовірнісних характеристик з визначеним математичним сподіванням та дисперсією входної ознаки. У кінцевому підсумку має бути сформовано уточнений початковий вектор первинних ознак, який забезпечить достовірне опрацювання нечітких даних.

Отже, опрацювання неповної або нечітко визначеної інформації полягає, з одного боку, у застосуванні теорії розмитої логіки (зокрема, теорії нечітких множин та мір) та побудові дерева логічного вислуду або, з іншого боку, у формуванні логічних правил, що містять функціонально-логічні залежності із невизначеними змінними, які можуть приймати як детерміновані, так і ймовірнісні значення. В процесі опрацювання апріорної інформації та виборі відповідних критеріїв правдоподібності можна поповнювати недостатні дані, забезпечуючи формування нових знань.

Для реалізації відповідних алгоритмів найбільш вдало пристосовані декларативні мови програмування, зокрема Refal, Reduce [11], Matcad [12] – мови аналітичних перетворень та системи автоматизації математичних вирахувань, Lisp, Skala [13] – аплікативні мови функціонального програмування, Prolog – мова вирахування логічних диз'юнктивів та правил [13, 14, 15].

Наведені підходи доцільно застосовувати у тих предметних галузях знань, для яких ускладнений або неможливий аналітичний опис засобами сучасної математики, або вимагаються колосальні технічні затрати і людські ресурси для повноцінного збору інформації про предметну область, зокрема, в галузі медицини, геології, екології, технічної діагностики, розпізнаванні та ідентифікації складно структурованих сигналів, процесів та явищ.

Література: 1. Уэно Х. Введение в инженерию знаний / Х. Уэно - Токио: Омся, 1985. 2. Новак В. Математические принципы нечеткой логики / В. Новак, И. Перфильева, И. Мочкорж. М: Физмалит, 2006. 347 с. 3. Zadeh L.A. Outline of a New Approach to the Analysis of Complex Systems and Decision Processes". IEEE. Trans. Syst. Van and Cybern, 1, 1973. P. 28-44. 4. Коньшева Л.К. Основы теории нечетких множеств / Л.К. Коньшева, Д.М. Назаров. М: Питер, 2011. 190 с. 5. Сявакко М. Математика прихованих можливостей / М. Сявакко. Острог: Видавництво НУ «Острозька академія», 2011. 394 с. 6. Турксен И.Б. Нечеткие экспертные системы / Под ред.

М. Желены. СПб: Питер, 2002. 1120 с. 7. *Шати́ро Д.И., Бли́шун А.Ф.* Выбор решений при нечетком состоянии системы. Алгоритмы и программы, 1978, №1, с.75. 8. *Zimmermann H.-J.* Fuzzy Set Theory and it's Applications. Kluner Academic Publishers, Dordrecht, Boston, MA, 2nd., 1991. 315 p. 9. *Дюбуа Д.* Теория возможности. Приложения к представлению знаний в информатике / Д. Дюбуа, А. Прад. М.: Радио и связь, 1990. 288 с. 10. *Ротштейн О.П.* Дифференциальная диагностика ишемической болезни сердца на основе нечеткой логики / Ротштейн О.П., Жупанова М.О., Шеверда В.М. Вестник ВПИ. №3. 1994. С. 32 -38. 11. *Еднерал В.Ф.* Язык аналитических вычислений REDUCE / В.Ф Еднерал, А.П. Крюков, А.Я. Родионов. М.: Из-во МГУ, 1984. 176 с. 12. *Дьяконов В.* Matcad 2001 // В. Дьяконов. СПб: Питер, 2001. 624 с. 13. *Заяць В.М.* Функційне програмування / В.М. Заяць. Львів.: Бескид Біт, 2003. 160 с. 14. *Ц. Ін.* Программирование в Турбо-Прологе / Ц. Ін, Д. Соломон. М.: Мир, 1996. 640 с. 15. *Заяць В.М.* Логічне та функціональне програмування. Навч. посібник. Львів: Українські технології, 2016. 400 с.

Transliterated bibliography:

1. *Ujeno H.* Vvedenie v inzheneriju znaniy / H. Ujeno. Tokio: Omsja, 1985.
2. *Novak V.* Matematicheskie principy nechetkoj logiki / V. Novak, I. Perfil'eva, I. Mochkorzh. M: Fizmalit, 2006.347 s.
3. *Zadeh L.A.* Outline of a New Approach to the Analysis of Complex Systems and Decision Processes". IEEE. Trans. Syst. Van and Cybern, 1, 1973. P. 28-44.
4. *Konyseva L.K.* Osnovy teorii nechetkih mnozhestv / L.K. Konyseva, D.M. Nazarov. M: Piter, 2011. 190 s.
5. *Sjavavko M.* Matematika prihovanih mozhlivostej / M. Sjavavko. Ostrog: NU «Ostroz'ka akademija», 2011. 394 s.
6. *Turksen I.B.* Nечеткие экспертные системы / Под ред. М. Желены. SPb: Piter, 2002. 1120 s.
7. *Shapiro D.I., Blishhun A.F.* Vybor reshenij pri nechetkom sostojanii sistemy. Algoritmy i programmy, 1978, №1, s.75.
8. *Zimmermann H.-J.* Fuzzy Set Theory and it's Applications. Kluner Academic Publishers, Dordrecht, Boston, MA, 2nd., 1991. 315 p.
9. *Djubua D.* Teorija vozmozhnosti. Prilozhenija k predstavleniju znaniy v informatike / D. Djubua, A. Prad. M.: Radio i svjaz', 1990. 288 s.
10. *Rotshtejn O.P.* Dyferencijna diagnostyka ishemichnoi' hvoroby sercja na osnovi nechitkoi' logiky / Rotshtejn O.P., Zhupanova M.O., Sheverda V.M. Visnyk VPI. №3. 1994. S. 32-38.
11. *Edneral V.F.* Jazyk analiticheskikh vychislenij REDUCE / V.F Edneral, A.P. Krjukov, A.Ja. Rodionov. M.: MGU, 1984. 176 s.
12. *D'jakonov V.* Matcad 2001 // V. D'jakonov. SPB: Piter, 2001. 624 s.
13. *Zajac' V.M.* Funkcijne programuvannja / V.M. Zajac'. L'viv.: Beskyd Bit, 2003. 160 s.
14. *C. In.* Programmirovanie v Turbo-Prologe / C. In, D. Solomon. M.: Mir, 1996. 640 s.
15. *Zajac' V.M.* Logichne ta funkcional'ne programuvannja. Navch. posibnyk. Gryf nadano MON Ukrai'ny (protokol № 1 vid 03.05.2013 r / V.M. Zajac', M.M. Zajac'. L'viv: Ukrai'ns'ki tehnologii', 2016. 400 s.

Надійшла до редколегії 25.10.2017

Рецензент: д-р фіз.-мат. наук, проф. Калинюк П.І.

Заяць Василь Михайлович, д-р техн. наук, професор Університету Технологічно-Природничого (м. Бидгощ, Польща), професор кафедри обчислювальної техніки Національного університету водного господарства (м. Рівне, Україна). Наукові інтереси: числові методи в прикладних застосуваннях, математичне і комп'ютерне моделювання динамічних систем, розпізнавання об'єктів зі складною динамічною природою, електрорадіотехніка та інформатика. Захоплення і хобі: туризм, велоспорт, театр; щоденні піші прогулянки 7 км. Адреса: Україна, 79070, Львів, пр. Червона Калина, 39 кв. 4, тел.: дом. (+38) (032) 222-43-48; роб.: (+48) 52-340-53-30; моб.: (+38)(032) 0678362473, e-mail: zvym01@rambler.ru

Рибницька Ольга Мар'янівна, канд. фіз.-мат. наук, доцент кафедри вищої математики НУ «Львівська політехніка». Наукові інтереси: застосування теорії нечітких множин та розмитої логіки до прикладних задач теорії управління та інформатики за галузями знань. Захоплення і хобі: подорожі, театр, література. Адреса: Україна, 79000, Львів, вул. С. Бандери, 23, роб. тел.: 258 22 50, дом.: 79038, Львів, вул. Китайська 75-А, тел. (032)2513327, e-mail: olga.rvbytska@gmail.com

Заяць Марія Михайлівна, ст. викладач НУ «Львівська політехніка». Наукові інтереси: інформаційні технології, інженерія знань, математичне і комп'ютерне моделювання складних об'єктів, логічне програмування. Захоплення і хобі: туризм, театр, якісне приготування вишуканих страв. Адрес: Україна, 79070, Львів, пр. Червона Калина, 39, кв. 4, тел.: дом. (+38) (032) 222-43-48; роб.: (+38) (032) 258-25-38 моб.: (+38)(032) 0989327526, e-mail: zajacmarija@gmail.com

Zaiats Vasyl M., DSc., Professor, Professor University of Science and Technology (m. Bydgoszcz, Poland). My research interests include numerical methods in applied applications, mathematical and computer modeling of dynamic systems, recognition of objects of complex dynamic nature, electro - radiotekhnika and informatics. Tastes and hobbies: hiking, biking, theater; daily walks 7 km. **Address, phones pin:** Ukraine, 79070, m. Lviv, avenue Chervona Kalyna №39 flat 4. Tel. : Home. (38) (032) 222-43-48; Job: (+48) 52-340-53-30; Mob.: (+38) (032) 0678362473.

Rybytska Olga Maryanivna, Ph.D., associate professor, associate professor of the department of higher mathematics, NU "Lviv Polytechnic". Scientific interests: application of the theory of fuzzy sets and blurry logic to applied problems of the theory of management and informatics by branches of knowledge. Address: 79000 m. Lviv, S. Bandera st., 23, work tel. 258 22 50, home address: 79038 Lviv, st. Chinese 75-A, house tel. (032) 251 33 27.

Zaiats Mary M., senior teacher National University "Lviv polytechnic" (Lviv, Ukraine). My research interests include information technology, engineering knowledge, mathematical and computer modeling of complex objects, logic programming. Interests and hobbies: hiking, theater, high-quality gourmet food. Address: Ukraine, 79070, m. Lviv, avenue Chervona Kalyna number 39 kv.4. Tel. (38032) 222-43-48; Job.: (+38) (032) 258-25-38 Mob.: (+38) (032) 0989327526.

ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ

УДК 519.7

ИССЛЕДОВАНИЕ МЕТОДОВ МНОГОКРИТЕРИАЛЬНОЙ ОЦЕНКИ УПРАВЛЕНИЯ ДЕЯТЕЛЬНОСТЬЮ СОВРЕМЕННОГО ПРЕДПРИЯТИЯ ГВОЗДИНСКИЙ А.Н., СЕРИК Е.Э.

Исследуются методы теории оптимизации, ориентированные на определение и идентификацию наилучших вариантов решения, не прибегая к проверке всех возможных решений путем реализации их на практике, а также методы многокритериальной оценки, позволяющие оптимизировать деятельность современного предприятия по нескольким критериям. Каждый из критериев является составляющим технологического процесса.

1. Введение

Одним из неперенных условий повышения эффективности функционирования предприятий в любой отрасли хозяйствования является комплексный и систематический анализ результатов их деятельности, направленный на выявление внутренних резервов производства, наращивание экономического потенциала и обеспечение требуемой рыночной конкурентоспособности. Особенно важным такой анализ является для тех производственно-хозяйственных структур, деятельность которых осуществляется в условиях резкого снижения спроса на их продукцию (услуги), высокой степени неопределенности получения государственных заказов и относительно невысокой в среднем финансовой состоятельности. Современный этап совершенствования системы управления народным хозяйством нашей страны основан на соединении преимуществ социалистического способа производства достижениями научно-технической революции. Результаты исследования задач планирования и управления показывают, что в реальной постановке эти задачи являются многокритериальными. Оценка деятельности предприятия и планирования как системы принятия решений производится на основе более десятка критериев: выполнение плана производства по объему, по номенклатуре, плана реализации, прибыли по показателям рентабельности, производительности труда. Особенно успешно развиваются методы оптимального планирования.

2. Задача и цели исследования

Цели данного исследования: изучение самых эффективных методов многокритериальной оценки, которые позволяют оптимизировать деятельность предприятия; оптимизация функционирования предприятия путем создания подсистемы поддержки принятия решений с использованием

информационных технологий, ЭВМ, системного программного обеспечения и математических моделей; создание системы, которая поможет определить наилучший и оптимальный вариант действий при наличии ограничений технико-экономического характера. При условии неполноты нужного количества ресурсов необходимо распределить их таким образом, чтобы наилучше удовлетворить всех потребителей и при этом получить максимальную прибыль или максимальный убыток. В работе проведена оценка классических и современных методологий решения многокритериальных задач, выявление их достоинств и недостатков, разработка и исследование математической модели, разработка методов и алгоритмов, позволяющих найти решение этой модели, создание программного продукта, спроектированного как полноценная система, способная решать любые оптимизационные задачи.

Задачей исследования является нахождение оптимального плана действия того или иного технологического процесса предприятия, который оценивается по нескольким критериям: максимизация прибыли и качества выпускаемой продукции, минимизация загрязнения окружающей среды.

Сущность исследований. Выбор метода решения – один из важнейших этапов оптимизации. Все многообразие современных методов подчас затрудняет выбор одного из них при решении различных оптимизационных задач. В работе рассмотрены наиболее продуктивные методы.

Проблема исследования. В связи с бурно развивающимся научно-техническим прогрессом возникает немало проблем. Наиболее важной является проблема совершенствования управления во всех звеньях народного хозяйства. Современные промышленные предприятия, научно-производственные и научно-исследовательские комплексы, комбинаты бытового обслуживания, т.е. самые разнообразные по характеру своей деятельности организации представляют собой сложные системы.

3. Описание и формализация задачи исследования

Целью работы является исследование деятельности предприятия в условиях рыночной экономики – максимизация прибыли или минимизация затрат. Затраты на производстве являются базой для установления цены.

Большинство задач, решаемых методами оптимизации, может быть сформулировано так: максимизировать или минимизировать $F(x_1, x_2, \dots, x_n)$ при ограничениях:

$$g_1(x_1, x_2, \dots, x_n) \leq b_1,$$

$$g_2(x_1, x_2, \dots, x_n) \leq b_2,$$

$$\dots\dots\dots$$

$$g_m((x_1, x_2, \dots, x_n) \leq b_m,$$

где $F(x_1, x_2, \dots, x_n)$ – целевая функция или критерий эффективности (например, прибыль, стоимость перевозок); $X = (x_1, x_2, \dots, x_n)$ – варьируемые параметры; $g(x) \dots g_m(x)$ – функции, которые задают ограничения на имеющиеся ресурсы. Именно при таких условиях и был промоделирован критерий максимизации прибыли для поставленной задачи.

Модель задачи квадратичного программирования имеет следующую структуру: найти максимальное значение функции показателя качества продукта

$$F_2(X) = C_t + 1/2 X^T D X =$$

$$= \sum_{j=1}^n C_j X_j + \frac{1}{2} \sum_{i=1}^m \sum_{j=1}^n d_{ij} x_i x_j \quad (1)$$

при нелинейных ограничениях:

$$a_{11}x_1 + a_{12}x_2 + \dots + a_{1n}x_n \leq b_1;$$

$$a_{21}x_1 + a_{22}x_2 + \dots + a_{2n}x_n \leq b_2;$$

$$\dots\dots\dots$$

$$a_{m1}x_1 + a_{m2}x_2 + \dots + a_{mn}x_n \leq b_m.$$

В связи с хозяйственной деятельностью объекта любой области в конкретном регионе непременно возникает природно-техническая система, которая включает как влияние на окружающую среду, так и изменение состояния. Задача минимизации загрязнения на производстве попадает под задачу минимизации функции (4) $F_3(X)$ при ограничениях. Параметры C_j неизвестны и зависят от характеристик оборудования и материальных ресурсов. Модель объекта представлена в виде:

$$AX \leq B, X \geq 0; \quad (3)$$

$$F_3(X) = \sum_j^n C_j x_j \rightarrow \min, \quad (4)$$

$F_3 = X_3 C^3 + E$, где F_3^t – вектор наблюдений; C^3 – вектор параметров, подлежащих оцениванию; E – вектор ошибок; $A = \{ij\}$, $i = 1, m$, $j = 1, n$ – матрица норм затрат ресурсов на единицу каждого вида продукции.

Ограничением на выпуск продукции различных типов служат продовольственные ресурсы $B = \{b_1, b_2, b_3, \dots, b_m\}$. С учетом норм затрат на единицу каждого вида продукции указанные ограничения можно записать в виде: выражение (3) описывает условия, которые необходимо учесть в годовой производственной программе, строкам матрицы A соответствуют все виды ре-

сурсов, рассматриваемые в задачах. Соответствующие строкам матрицы A компоненты вектора B указывают ограничения видов ресурсов или объектов производства, которые установлены для годовой производственной программы предприятия. Эти неравенства представляют собой обычные условия неотрицательности.

Общая постановка задачи состоит в том, чтобы определить вектор $X(0)$, обеспечивающий компромисс между величиной прибыли, качеством выпускаемой продукции и уровнем загрязнения окружающей среды.

Один из возможных методов решения состоит в том, что вначале находятся три оптимальных вектора производства, каждый из которых соответствует одному из локальных критериев. Затем определяется выпуклая линейная комбинация $X(0)$, представляющая собой оптимальную программу относительно указанных критериев:

$$X^{(0)} = v_1 x^{(1)} + v_2 x^{(2)} + v_3 x^{(3)}. \quad (5)$$

Для нахождения v_1, v_2, v_3 используется игровая модель, для решения которой был выбран матричный метод.

4. Анализ методов исследования

Математическая модель – это сложная система: с распределенными и сосредоточенными параметрами, линейными и нелинейными, стохастическими и детерминированными. Параметры модели зачастую непрерывные. Область поиска может быть несвязной, необходимая информация отсутствовать, функции цели могут быть недифференцируемыми, а размерность векторов критериев и параметров может достигать огромной величины.

Основное преимущество метода в том, что во время производства, в ходе анализа пространства параметров поступает важная информация о качестве математической модели. Её корректируют или, если это необходимо, строят новую. Так, метод выступает как индикатор достоверности модели с точки зрения позиций исследуемых показателей.

Многокритериальность – это главное свойство. Большинство авторов старается свести многокритериальные задачи к однокритериальным. При однокритериальной постановке задачи без ответа остаются важные вопросы: какой ценой это достигается, в какой мере ухудшаются при этом другие критерии и почему мы отдаем предпочтение этому, а не другому критерию, стараясь свести многокритериальную задачу к другой, которая очень отличается от начальной.

5. Программная реализация разработанной модели

Для программной реализации математической модели может быть использовано программное обеспечение, позволяющее проводить оценива-

ние по каждому из описанных критериев. Логика программы основана на симплекс-методе. Его идея заключается в последовательном продвижении по базисам опорных планов вплоть до получения оптимального решения или доказательства неразрешимости задачи. Значение целевой функции при этом увеличивается, а система ограничений приводится к каноническому виду. По найденной целевой функции и системе ограничений строится симплекс-таблица. Данные на выход подаются нормализованными и представленными в каноническом виде, т.е. предварительно обработанными.

Следует отметить, что использованный в работе симплекс-метод решения оптимизационных задач многокритериального типа не является единственным. В настоящее время для решения задач принятия решений при оценке деятельности предприятия могут быть использованы эволюционные методы оптимизации.

6. Выводы

Проведена оценка классических и современных методологий решения многокритериальных задач, выявлены их достоинства и недостатки, разработана математическая модель, методы и алгоритмы, позволяющие найти решение указанной выше модели, создан программный продукт, спроектированный как система, способная решать оптимизационные задачи, которые разрешимы разработанными методами и алгоритмами. *Научной новизной* результатов исследования, приведенных в работе, является метод решения полученной математической модели, наиболее полно охватывающий её на базе теории оптимизации.

Принцип решения полученной модели наиболее соответствует традиционным методам теории оптимизации. Метод решения разработанной математической модели основан на использовании методов теории игр. Он легко реализуется программно и позволяет найти оптимальное решение – один из вариантов решения проблемы, имеющий наилучшую оценку.

Литература: 1.Батыщев Д.И. Методы оптимального проектирования М.: Радиосвязь, 1984. 246 с. 2.Гвоздинский А.Н., Губин В.А., Шергин В.Л. Методы оптимизации в организационном управлении. Х.: ХНУРЭ. 2014.396 с.3. Лэсдон Л.С. Оптимизация больших систем. М: Наука, 1975, С.65-83. 4.Гвоздинский А.Н. Принципы и методы оптимизации в современных организационных системах управления: Научно-технический журнал [Текст] / А.Н.Гвоздинский, А.Е.Козлова, А.О.Дроздов // Х: Бионика интеллекта, 2013. №2 (81). С.66-70. 5.Гвоздинский А.Н. Методы аналитической обработки информации [Текст] / А.Н. Гвоздинский, Е.Г. Климко // Радиоэлектроника и информатика. 2000. №4. С.111-112.

Transliterated bibliography:

1. Batyshhev D.I. Metody optimal'nogo proektirovanija M.: Radiosvjaz', 1984. 246 s. 2. Gvozdinskij A.N., Gubin V.A., Shergin V.L. Metody optimizacii v organizacionnom upravlennii. H.: HNURJe 2014. 396 s. 3.Ljesdon L.S. Optimizacija bol'shih sistem. M: Nauka, 1975, S.65-83. 4. Gvozdinskij A.N. Principy i metody optimizacii v sovremennyh organizacionnyh sistemah upravlenija: Nauchno-tehnicheskij zhurnal [Tekst] / A.N. Gvozdinskij, A.E.Kozlova, A.O.Drozdoz // H: Bionika intellekta, 2013. №2 (81). S. 66-70. 5. Gvozdinskij A.N. Metody analiticheskoy obrabotki informacii [Tekst] / A.N. Gvozdinskij, E.G. Klimko // Radiojelektronika i informatika. 2000. №4. S.111-112.

Надійшла до редколегії 17.05.2017

Рецензент: д-р техн.наук, проф. Путятин Е.П. Гвоздинский Анатолий Николаевич, канд. техн. наук, профессор кафедры искусственного интеллекта ХНУРЭ. Научные интересы: оптимизация процедур принятия решения в сложных системах управления. Адрес: Украина, 61166, Харьков, ул. Академика Ляпунова, 7, кв. 9, тел. 702-38-23.

Серик Екатерина Эдуардовна, студентка группы КН-13-6 кафедры искусственного интеллекта ХНУРЭ. Научные интересы: методы принятия решения в системах искусственного интеллекта. Адрес: Украина, 61000, Харьков, пер. Афанасьевский, дом 36, тел. 066-340-08-76.

Предлагаются структуры киберсоциального компьютинга, которые рассматриваются как компоненты киберфизических технологий для мониторинга и управления обществом. Излагаются основные тренды развития киберфизического уклада, представленные в Gartner's Hype Cycle 2017 в целях их применения в науке, образовании, транспорте, индустрии и государственных структурах. Предлагаются перспективные направления рыночно привлекательных технологий, которые касаются киберсоциального мониторинга и управления обществом. Формируется расширенное описание технологий, связанных с умным цифровым миром, зелеными городами и 5G-телекоммуникациями. Даются рекомендации к использованию топ 10 технологий 2017 года в бизнесе и научно-образовательных процессах высшей школы. Формулируется D-метрика измерения расстояний между процессами и явлениями в киберфизическом пространстве в виде универсальной формулы для точного определения дуги обратного транзитивного замыкания по двум известным расстояниям, дополняющим замыкание до конволюционного цикла.

Ключевые слова: hype cycle emerging technologies, digital world, cyber-physical computing, machine learning, metric of spaces, smart digital state, memory-driven computing, logical and emotional computing, cyber social computing.

1. Введение

Ключ в будущее человечества содержит пять компонентов: 1) Цифровой мир. 2) Новые материалы. 3) Генная инженерия. 4) Мутация природы. 5) Солнце-энергия. Наше будущее сегодня связывается не столько с научно-техническим прогрессом, сколько с сохранением зеленой экологии планеты. Природа имеет собственный геном или алгоритм развития, который предполагает существование начала и конца. Геном развития имеют также все компоненты природы: Вселенная и ее космологические субъекты, земная природа, включающая живые существа, физические объекты (естественные и искусственные) и человечество. Для исполнения собственного генома человечество, как часть природы, имеет мощные механизмы саморегулирования или иммунитета, которые предназначены для сохранения жизни на планете путем борьбы с многочисленными социальными вирусами. По мере развития человечества локальные социальные вирусы (коллизии и катаклизмы, революции и войны) мутируют в совершенство, приобретая все более опасные глобальные формы (национализм, нацизм, терроризм, коррупционизм), спо-

собные уничтожить живые формы жизни на планете. Носителем социального вируса является социальная группа, в которой отсутствует элементарная гигиена поддержания нравственных отношений, прописанных в библии. При отсутствии иммунитета и вакцинации социальный вирус сегодня практически мгновенно способен поражать страны и континенты в результате воздействия интернет-коммуникаций. Что же такое социальный вирус? Негативное отношение к дифференциалу (разности) морально-материального состояния двух социальных субъектов (рис. 1), где N и M – соседи, P(N) и P(M) – их собственности (состояния).

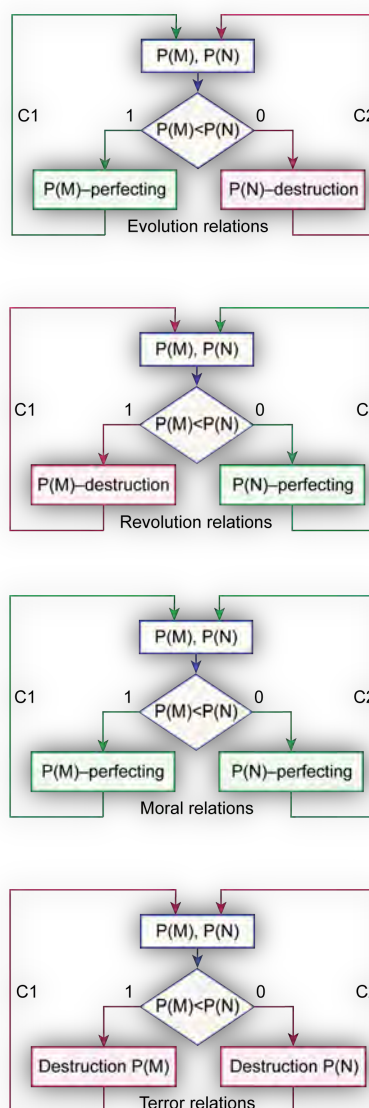


Рис. 1. Отношения между двумя социальными субъектами

Существует четыре масштабируемых типа отношений между двумя социальными субъектами M (от первого лица) и N (соседи), проектируемых на множество людей, компаний и государств: 1) Я – эволюционер. Здесь эволюцион-

ные отношения основаны на моем совершенствовании P(M), нацеленном на разрушение соседа P(N). Становитесь сильным и уничтожайте слабых. 2) Я – революционер. В этом случае революционные отношения основаны на разрушении моего достояния P(M), нацеленном на совершенствование чужого P(N). Уничтожайте все свое наследие (историю, культуру, языки, экономику) и создавайте новое по чужим шаблонам. 3) Я – моралист. При этом моральные отношения основаны на моем совершенствовании P(M), нацеленном на последующую помощь соседу P(N). Сделайте успешным себя и помогите ближнему. 4) Я – террорист. Здесь террористические отношения основаны на моей самодеструкции P(M), нацеленной на еще большее разрушение соседей P(N). Убей себя вместе с окружающим миром. Существует только один тип моральных отношений с точки зрения представленного алгоритма: «Факт – Оценка – Действие», когда активность выполняется под лозунгом: «Мое совершенствование нацелено на помощь людям». Остальные три типа отношений, упомянутых выше, являются аморальными.

Другая интерпретация. Если сосед живет лучше, то существует два пути развития субъекта: 1) Позитивный, но длительный во времени – познать опыт соседа и превзойти его достижения честным творческим трудом. 2) Негативный, но самый эффективный по времени – уничтожить соседа и/или украсть его собственность.

Что же такое социальный иммунитет? Нравственные отношения между людьми, созданные конституцией, культурой и традициями, стимулирующие познание опыта человечества и доминирование творческого совершенства гражданина в социальной значимости его труда. Такие нравственно здоровые отношения в обществе не могут быть поражены социальными вирусами национализма, терроризма и коррупционизма. Альтернативой выступает известная доктрина социального равенства всех граждан в доходах, которая обрекает социальную группу или государство на деградацию: «Мы одинаково счастливы в равенстве своей нищеты». Тем не менее, полное исполнение генома человечества вполне возможно при условии создания политической элитой планеты социального иммунитета – глобальных нравственных отношений, направленных на обеспечение качества жизни каждого человека и сохранение экологии планеты. Одним из возможных вариантов такого иммунитета является киберсоциальный компьютеринг, призванный поддерживать гигиену социальных отношений, предотвращающую проникновение социальных вирусов в умы и сердца людей.

Целью исследования является разработка логических компонентов киберсоциального компьютеринга на основе нравственных, метрических социальных отношений для точного облачного управления социальными процессами, использования больших данных, онлайн-мониторинга мнения социальных групп в целях улучшения качества жизни человека и сохранения зеленой экологии планеты.

Задачи: 1) Определение структуры зеленого управления социальными группами на основе нравственных общественных отношений для предотвращения социальных коллизий и катаклизмов. 2) Анализ современных киберфизических трендов развития зеленых технологий управления физическими и социальными процессами, представленных компанией Gartner. 3) Практика использования зеленых топ-технологий киберкультуры 2017 для университетов, компаний и государств. 4) Эмоционально-логический компьютеринг: структура и компоненты. 5) Моделирование социальных процессов для принятия решений и прогнозирования их последствий. 6) Киберфизическая государственность, как будущее устройство зеленого социального мира. 7) D-метрика измерения расстояний между социальными процессами и явлениями в киберфизическом пространстве.

Киберсоциальный компьютеринг представляет собой теорию и практику нравственных, социальных отношений для точного облачного управления виртуальными и социальными процессами и явлениями на основе использования больших данных, метрического онлайн-мониторинга мнения социальных групп в целях улучшения качества жизни человека и сохранения экологии планеты.

Существуют определенные тенденции в мире, которые являются технологической основой для создания киберсоциального компьютеринга, как части киберфизического компьютеринга, в рамках технологического уклада Internet of Things. Компания Gartner Inc., предсказывающая глобальную технологическую кибермоду, в 2017 году добавила восемь новых трендов в свой бренд Hype Emerging Technologies Cycle: 5G, Artificial General Intelligence, Deep Learning, Deep Reinforcement Learning, Digital Twin, Edge Computing, Serverless PaaS, and Cognitive Computing [14].

2. Эмоционально-логический компьютеринг (EL-computing)

“Спокойная и скромная жизнь приносит больше счастья, чем погоня за успехом, которая вызывает лишь постоянный стресс” (Альберт Эйнштейн). “Все есть слово и слово есть Бог” (Евангелие от Иоанна). Слово создает и разрушает, возносит и ниспровергает, лечит и калечит, опо-

вещает и управляет, делает богатым и нищим, мудрым и глупым. Для компьютеринга слово есть: команда и ресурс, состояние и цель, управление и оповещение, логика и эмоции, создающие архитектуру. Библия не только интегрирует историю или опыт человечества (оповестительные сигналы), но и программирует каждого человека на достойное поведение путем выработки оперативных регуляторных воздействий, имея в виду функцию цели – нравственное развитие человечества во имя спасения планеты. Другими словами, библия есть первая и пока самая успешная попытка создания эмоционально-логического компьютеринга в форме искусственного интеллекта человечества, где на любой вопрос, связанный с принятием решения, каждый человек может найти практически точный ответ. Эмоции и логика связаны с функционированием двух сопроцессоров: правого и левого полушария головного мозга, которые участвуют в мониторинге ситуации и/или окружающей действительности для принятия решения в направлении движения к наперед заданной цели. Полушария головного мозга есть не что иное, как два взаимодополняющих альтернативных сопроцессора, к которым имеет доступ сам человек и/или другие люди. Взаимодействие полушарий для выработки решения укладывается в четыре формы, представленные на рис. 2. Типы эмоционально-логического компьютеринга можно обозначать символами алфавита Кантора: $\{0, 1, X, \emptyset\}$. Здесь фигурируют два полушария $\{0, 1\}$ головного мозга, которые создают следующие типы эмоционально-логического компьютеринга (EL-computing): 1) Логический (0), когда развитое левое полушарие доминирует в процессе принятия решения, основанного на приобретенных знаниях и богатом жизненном опыте, характерном для немолодых и/или нетворческих людей. 2) Эмоциональный (1), когда развитое правое полушарие доминирует в процессе принятия решения, основанного на эвристических озарениях и эмоциональном опыте, характерном для молодых и/или творческих людей. 3) Эмоционально-логический ($X=\{0, 1\}$), когда одинаково развитые полушария мозга участвуют в процессе принятия решения, основанного на приобретенных знаниях и эмоциональном опыте, характерном для образованных и творчески одаренных людей. 4) Повторительный ($\emptyset$), когда одинаково неразвитые полушария мозга, не способные принимать самостоятельные решения ввиду отсутствия знаний и опыта, исполняют функцию повторителя, что характерно для необразованных и/или слабо-

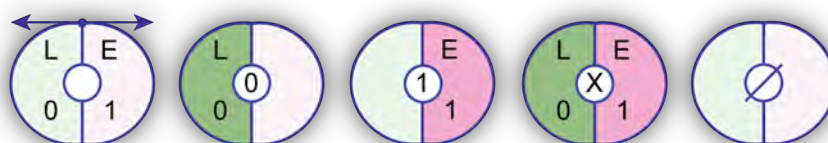


Рис. 2. Типы эмоционально-логического компьютеринга

вольных людей. Интересно, что практически любой образовательный процесс (детский сад, школа, университет, предприятие) направлен на повышение уровня логического интеллекта за счет уменьшения уровня эмоционального интеллекта, связанного с творческой деятельностью. Знания помогают избежать тривиальных ошибок, допущенных человечеством, но ограничи-

вают полет фантазии для открытия новых технологий, материалов, процессов и явлений.

Emotional logical (psychological) computing. Эмоционально-логический (психологический) компьютеринг есть метрическое управление эмоциональным и логическим сопроцессорами головного мозга на основе цифрового мониторинга киберсоциальной среды для выбора оптимального решения на пути достижения поставленной цели (рис. 3).

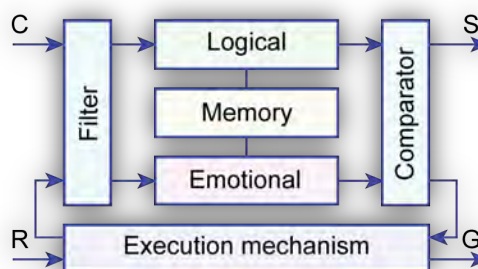


Рис. 3. Модель эмоционально-логического компьютеринга

Объективно следует рассматривать, что такой компьютеринг является доступным как со стороны владельца мозга, так и со стороны окружающих его людей, с позитивными и негативными целями. Такой дуализм доступа к сопроцессорам мозга имеет свои плюсы и минусы: 1) Хорошо, что я имею влияние не только на себя, но и на других людей, чтобы исцелить их, сделать успешными и счастливыми. 2) Плохо, что, зная структуру сопроцессоров и коды доступа, можно легко перепрограммировать любого гражданина на безнравственные поступки для нанесения вреда ему, социальным группам и экологии планеты. Программировать поведение человека означает включать функциональные модули его эмоционального и/или логического интеллекта (сoproцессора) такими оповестительными сигналами, которые приведут к желаемому, собственному

или навязанному кем-то решению. Иначе, человек может сам себя программировать, а также это могут делать и другие люди для достижения своих целей. Если логические процессоры принятия решений достаточно описаны в современной научной литературе [11], то создание адекватных эмоциональных процессоров находится на стадии разработки и верификации. Однако обращает на себя внимание одно знаменательное событие – присуждение Нобелевской премии за 2017 г. Ричарду Талеру (Richard Thaler), ученому, работающему в области финансово-экономического поведения граждан. Он создал «теорию подталкивания или управляемого выбора», основанного на постепенном переходе от эмоций к логике принятия решения. Тем не менее, в большинстве своем человечество пока руководствуется, как раз наоборот, эмоциями для выработки линии поведения. Например, желание не потерять того, что имеешь, для народных масс является более предпочтительным, чем возможное приобретение большего.

Эмоциональный процессор, помимо памяти для хранения данных, имеет функционально полный базис для описания примитивов, схем и устройств. Эмоциональный процессор оперирует ярко выраженными позитивными и негативными аксиомами, высказываниями и действиями. Негатив более популярен в бытовой речи, чем позитив. Например, слово “умник” имеет 18 синонимов, а “дурак” – 192 синонима. Однако при анализе 100 самых популярных печатных слов в русском языке выяснилось, что только одно из них – “плохой” носит явно негативный оттенок. Слово “логика” проигрывает перед словом “эмоция” в количестве синонимов как: 4 – 27 [http://synonymonline.ru]. Иначе, принятие решения, основанное на негативных эмоциях толпы, направленных против конкретных людей, процессов или явлений, является беспроигрышной технологией прихода к власти. В мудрости мы всегда находим мизерные недостатки, а глупость украшаем раздутыми достоинствами.

Для создания базиса примитивов Е-компьютинга предлагается аксиоматическая метрика эмоционального поведения человека, построенная на доминировании минимализма (самого короткого пути) в любой системе отношений, которая присуща социальным массам, не имеющим специального логического образования. Для аналитического задания отношений на ориентированном графе используется Алгебраическая форма представления графа (АФПГ) [12]. Метрика эмоциональных отношений или Е-компьютинга представлена тремя аксиомами:

1) Рефлексивность $(a \rightarrow a)$: Самый короткий путь в системе отношений. Замкнутая на себя система сохраняет свое и не приобретает чужого.

2) Симметричность $[(a \rightarrow b) \vee (b \rightarrow a)]$: Минимум связей с внешним миром. Я даю тебе, ты возвращаешь мне. Толерантность в отношениях и отсутствие посредников.

3) Транзитивность $[(a \rightarrow b) \vee (b \rightarrow c) \vee (a \rightarrow c)]$: Минимальный путь к цели. Прямой контакт с субъектом, исключая посредника. Убрать лишнее звено в цепи событий.

Выполнение данных аксиом, создающих поведенческую метрику минимализма, формирует в социуме бинарные отношения эквивалентности, что есть хорошо для системы в целом (рис.4):

$$Y = aa \vee bb \vee cc \vee ab \vee ba \vee ac \vee ca \vee bc \vee cb.$$

Однако каждый индивидум имеет эгоистическое стремление к модификации системы толерантных эквивалентных отношений под себя, превращая их в нерелексивные, несимметричные и нетранзитивные (рис. 4, c- and r-graph).

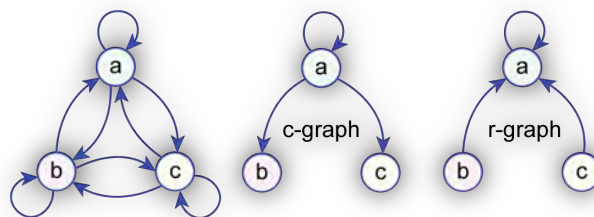


Рис. 4. Эквивалентные и эгоистические отношения

Эгоизм определяется структурой, минимизирующей метрику связей в системе отношений для одной вершины (субъекта): $Y^c = aa \vee ab \vee ac$ – авторитарное управление; $Y^r = aa \vee ba \vee ca$ – восходящие финансово-ресурсные потоки. Масштабируемое обобщение такой модификации порождает национализм – ненависть к другим народам, а также нацизм, где эквивалентные отношения трансформируются в пирамиду с титульной нацией на вершине, которая законодательно доминирует над второсортными народами.

Тем не менее, эгоистические отношения являются доминирующими в народных массах, которые ориентированы на быстрый, локальный и минимальный успех каждого человека без риска потерь и невозможность получения через длительное время больших ресурсных выигрышей. Естественно, что отношения эквивалентности по определению означают юридическое равенство всех граждан, дающее возможность в полной мере участвовать в производственных, экономических, политических и культурных процессах социальной группы с последующим получением

вознаграждения в соответствии с измеренной социальной значимостью результатов труда каждого человека. Эквивалентные отношения привлекают в социальную группу лучших специалистов, которые производят конкурентоспособную продукцию и сервисы для успешной продажи на рынке, что приносит высокую прибыль, которая обеспечивает зарплату с большим числом знаков, которая стимулирует человека к более производительной творческой деятельности.

Метрика эмоциональных отношений по горизонтальным и вертикальным связям порождает “правила” поведения среднестатистического человека: 1) Сохранять свое лучше возможного приобретения чужого. 2) Нарушать правила при гарантированном отсутствии наказания. 3) Маленькая зарплата сегодня лучше обещанных тысяч завтра. 4) Избирать и уважать тех, у кого реальная власть. 5) Заботиться о себе, а потом о других. 6) Слушать себя и подчиняться руководителям, игнорировать мнения подчиненных. 7) Получать удовольствие от недостатков и ошибок других людей. 8) Обсуждать третьих лиц, хвастаться успехами и жаловаться на судьбу. 9) Вступать в эмоциональные конфликты с окружающими. 10) Не выполнять обещания и не возвращать долги. 11) Становиться в ряды большинства, которое объединилось против человека или субъекта. 12) Не высказывать собственного мнения, если оно не совпадает с мнением руководителя или большинства. 13) Доверительно относиться к государственной пропаганде, идущей от властных структур.

Интерес представляет компьютерный алгоритм циклически бесконечной творческой деятельности субъекта: человека, ученого, руководителя, цель которого - приобретение социальной и материальной значимости результатов собственного труда, рис. 5.

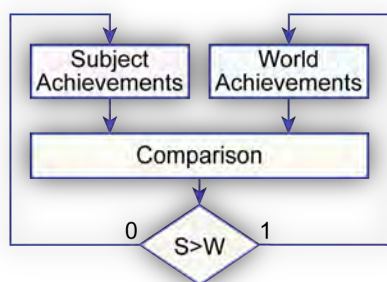


Рис. 5. Алгоритм творческого процесса

Источником для вдохновения являются мировые достижения, с одной стороны, и критическое отношение к конкретным видам продукции и сервисов - с другой. Если достижения субъекта выше, чем мировые аналоги, то его результаты через их социальное признание пополняют банк мировых достижений. В противном случае субъект

циклически совершенствует свои достижения: разработки, продукты или сервисы до уровня, когда они станут лучше мировых аналогов и/или пригодными для продажи на международном рынке. Предложенный алгоритм масштабируется на каждого человека, компанию, университет, организацию и государство.

3. Моделирование социальных процессов

Эмоциональная логика использует базис примитивных функциональностей: and, or, not, хог, которых достаточно для моделирования и предсказания поведения человека или социальной группы. Функционирование социальной системы можно представить в виде логического выражения: $Y = f(X)$, где X – конечное множество факторов, воздействующих на систему для формирования ее состояния. Далее используется аппарат булевых производных, который связывает изменение состояния входной переменной X с изменением состояния социальной системы Y :

$$\frac{df}{dx_i} = f(x_1, x_2, \dots, \bar{x}_i, \dots, x_n) \oplus f(x_1, x_2, \dots, x_i, \dots, x_n) = f(x_1, x_2, \dots, 0, \dots, x_n) \oplus f(x_1, x_2, \dots, 1, \dots, x_n).$$

Факторы или переменные, при активизации социальных процессов, делятся на существенные (а), обеспечивающие (с) и избыточные (n):

$$x_i \in \{X^a, X^c, X^n\}; X^a \cap X^c \cap X^n = \emptyset; X^a \cup X^c \cup X^n = X.$$

Существенная переменная изменяет или активизирует состояние социальной системы при определенных значениях остальных переменных, которые выполняют роль условий обеспечения социального процесса:

$$\frac{df}{dx_i} = f(x_1, x_2, \dots, \bar{x}_i, \dots, x_n) \oplus f(x_1, x_2, \dots, x_i, \dots, x_n) \neq 0.$$

Переменная является несущественной, если не существует входного условия для активации состояния выхода Y с помощью данной переменной:

$$\frac{df}{dx_i} = f(x_1, x_2, \dots, \bar{x}_i, \dots, x_n) \oplus f(x_1, x_2, \dots, x_i, \dots, x_n) = 0.$$

Вычисление булевой производной, которая в общем случае является функцией от (n-1) переменной, дает возможность определить условия для изменения состояния социальной системы при воздействии на нее изменением переменной (фактором, идеей) или задекларировать, что таких условий не существует, поскольку фактор является несущественным и не оказывает влияния на социум. Сказанное можно продемонстрировать на следующих примерах.

Пример 1. Пусть имеется система принятия судьбоносных государственных решений:

$$Y = f(P, L, R, U, B, M, V),$$

которая зависит от существенных переменных, представленных следующими субъектами: президент P , парламент L , правительство R , прези-

дент США U, олигархи В, военные М, народ V. Несложно составить логическую схему управления социумом, которая имеет всего три возможных варианта:

$$Y = PLR \vee BUV \vee MBU.$$

Взятие производной по каждой переменной позволит определить условия для реализации идеи в стране, инициируемой каждым социальным компонентом. Например, условия для реализации воли президента определяются следующей булевой производной:

$$\begin{aligned} \frac{df}{dP} &= (0LR \vee BUV \vee MBU) \oplus (1LR \vee BUV \vee MBU) = \\ &= (BUV \vee MBU) \oplus (LR \vee BUV \vee MBU) = \\ &= [BU(V \vee M)] \oplus [LR \vee BU(V \vee M)] = \\ &= [BU(V \vee M)] \oplus [LR \oplus BU(V \vee M) \oplus LRBU(V \vee M)] = \\ &= BU(V \vee M) \oplus LR \oplus BU(V \vee M) \oplus LRBU(V \vee M) = \\ &= LR \oplus LRBU(V \vee M) = \\ &= LRBU(V \vee M) = LR(\bar{B} \vee \bar{U} \vee \bar{V} \vee M) = \\ &= LR(\bar{B} \vee \bar{U} \vee \bar{V} \vee M) = \\ &= LR\bar{B} \vee LR\bar{U} \vee LR\bar{V} \vee LR\bar{M}. \end{aligned}$$

Таким образом, чтобы идея, инициируемая президентом, изменила страну, необходимо выполнение одного из трех условий, полученных путем взятия производной от функции $Y=f(X)$ управления социумом:

$$\frac{df}{dP} = LR\bar{B} \vee LR\bar{U} \vee LR\bar{V} \vee LR\bar{M}.$$

Структурная схема, представленная на рис. 6, имеет логические элементы для принятия решений, путь активизации идеи (воли), а также три условия активизации переменной Р для изменения выхода социальной структуры или состояния государства: 1) парламент, правительство, бизнес – нейтралитет; 2) парламент, правительство, президент США – нейтралитет; 3) парламент, правительство, народ – нейтралитет, военные – нейтралитет.

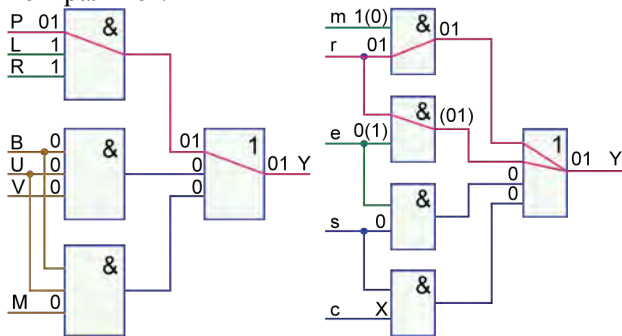


Рис. 6. Логические схемы управления: государством и университетом

Пример 2. Пусть имеется логическая структура управления университетом

$$Y = f(m, r, e, s, c, u),$$

которая включает следующие переменные соответственно: министр, ректор, ректорат, ученый совет, конференция представителей трудового

коллектива, коллектив университета. Логическая схема управления университетом (см. рис. 6) при имплементации идеи имеет следующие варианты реализации:

$$Y = mr \vee re \vee es \vee sc.$$

Пояснения: 1) Реализация идеи путем издания приказа ректора, согласованного с министром. 2) Реализация идеи путем издания приказа, согласованного с ректоратом. 3) Активизация идеи путем ее инициирования ректоратом и ученым советом. 4) Актуализация идеи путем ее инициирования конференцией трудового коллектива совместно с ученым советом.

Взятие булевой производной, например, по переменной ректора, определяет все возможные условия для актуализации идеи в университете, инициируемой первым руководителем:

$$\begin{aligned} \frac{df}{dr} &= (m0 \vee 0e \vee es \vee sc) \oplus (m1 \vee 1e \vee es \vee sc) = \\ &= (es \vee sc) \oplus (m \vee e \vee es \vee sc) = \\ &= (es \vee sc) \oplus (m \vee e) \vee (es \vee sc) = \\ &= (es \vee sc) \oplus (m \vee e) \oplus (es \vee sc) \oplus (m \vee e)(es \vee sc) = \\ &= (m \vee e) \oplus (m \vee e)(es \vee sc) = \\ &= (m \vee e)(\overline{es \vee sc}) = (m \vee e)(\bar{e}\bar{s}) = \\ &= (m \vee e)(\bar{e} \vee \bar{s})(\bar{s} \vee \bar{c}) = \\ &= m\bar{e}\bar{s} \vee e\bar{e}\bar{s} \vee m\bar{s}\bar{s} \vee e\bar{s}\bar{s} \vee m\bar{e}\bar{c} \vee e\bar{e}\bar{c} \vee m\bar{s}\bar{c} \vee e\bar{s}\bar{c} = \\ &= m\bar{e}\bar{s} \vee m\bar{s} \vee e\bar{s} \vee m\bar{s}\bar{c} \vee e\bar{s}\bar{c} = m\bar{s} \vee e\bar{s}. \end{aligned}$$

Таким образом, условия для активизации решения ректора представлены двумя вариантами событий, которые позволяют имплементировать идею в жизнь:

$$\frac{df}{dr} = m\bar{s} \vee e\bar{s} = (m \vee e)\bar{s}.$$

Структурная схема для активизации идеи (воли) ректора имеет условия, которые определяются согласием министра или ректората при непротивлении ученого совета (см. рис. 6).

Резюме: проектирование логической схемы управления социальными группами основано на формализации вербального описания системы власти. Имея построенную логическую схему менеджмента, достаточно просто определить условия для реализации любой идеи, идущей от существенных субъектов (переменных), путем взятия булевой производной. Предложенные модели теоретически объясняют социальные процессы, но на практике все выглядит значительно проще: “Буржуазная демократия есть скрытая форма диктатуры капитала” (В.И. Ленин). Ничего не изменилось за последние сто лет. Сегодня это актуально звучит так: “Демократия есть скрытая форма диктатуры олигархов”.

4. Киберфизическая государственность

“Где есть воля, есть и путь” (Альберт Эйнштейн). Киберфизическая модель государственности предназначена для метрического управле-

ния ресурсами и гражданами на основе цифрового мониторинга и оценивания потребностей социальных групп в целях обеспечения качества их жизни и процветания страны путем производства и экспорта товаров и услуг (рис. 7). Существует 6 основных системообразующих компонентов для функционирования государственности, упорядоченных по степени их влияния на рыночный успех проекта: 1) Отношения. 2) Цели. 3) Кадры. 4) Управление. 5) Инфраструктура. 6) Ресурсы.

1) Отношения. Государственность создается, прежде всего, нравственными метрическими цифровыми отношениями между гражданами, которые формируются конституцией, законами, положениями, приказами, бережным и толерантным отношением к языковой культуре, истории и традициям, объединяющим граждан. Политическая деструктивная элита часто формирует отношения, разъединяющие общество по вопросам языков, культуры и истории в целях длительного и тотального владычества над искусственно разрозненными социальными группами, не ориентированными на создание рыночно востребованной продукции.

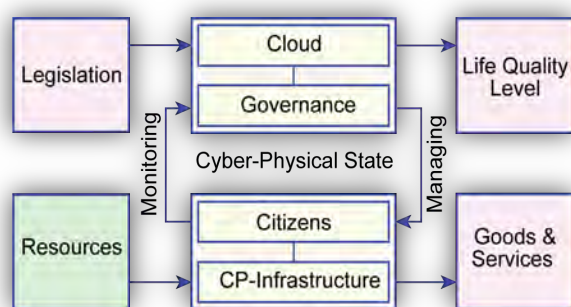


Рис. 7. Модель киберфизической государственности

2) Цели. Так или иначе, отношения формируются политической элитой, исходя из целей или доктрины государства, которая в цивилизованных странах направлена на нравственное развитие гражданина, рост его благосостояния и сохранение экологии за счет ресурсов, получаемых от экспорта товаров и услуг. Невежественность политической элиты в части создания механизма для производства товаров и услуг способна создать привлекательную доктрину войны и ненависти к другим народам и государствам для повышения качества жизни титульной нации за счет деструкции внутренних и внешних врагов.

3) Кадры. Естественно, товары и услуги создаются компетентными кадрами, которые должны быть правильно расставлены в иерархической структуре отношений «управление – исполнение» для достижения максимальной производительности труда при создании рыночно-востребованных товаров и услуг. Привлече-

ние лучших кадров из всех стран мира должно быть возведено в ранг государственной политики путем создания соответствующих моральных и материальных условий для сервисного обслуживания специалистов.

4) Управление. При этом аппарат (облачного) управления (с дипломами MBA), как ключевой компонент успеха, не должен пересекаться с исполнительным механизмом и составлять не более 10 процентов кадрового состава в целях минимизации накладных расходов на аналогичном уровне. Непроизводительные затраты в современном компьютере занимают сегодня не более 5 процентов, что является образцом для подражания в процессах управления государством. В прошлом иерархический аппарат управления достигал 50 процентов при отсутствии эффективных средств коммуникаций. В настоящее время прямые киберфизические контакты в Internet обеспечивают возможность облачного оперативного online управления всеми гражданами страны одновременно. Чиновниками должны быть только эксперты, получившие специальное образование по MBA, равно как 90 процентов всех оцифрованных информационных потоков должно быть трансформировано в электронный документооборот на основе облачного управления.

5) Инфраструктура. Становится все более доминирующей киберфизическая инфраструктура государства, которая заставляет политическую элиту образовываться в части использования новых технологий для безбумажного цифрового управления гражданами. Роль чисто физической инфраструктуры трансформируется во второстепенные символы государственности, связанные с границей, дорогами и зданиями, создающими комфортные условия для каждого гражданина планеты. Здесь доминирующей технологией становится IoT, которая создает е-инфраструктуру для авторизованного online доступа к киберфизическим компонентам государства на основе использования первичных признаков гражданина (скан пальцев, лица или сетчатки глаза).

6) Ресурсы. Они ставят в функциональную зависимость от перечисленных выше компонентов, даже при изначальном нулевом бюджете нового государства. Свободных денег в мире больше, чем интересных проектов, направленных в позитивное будущее. Чтобы найти спонсоров, достаточно продемонстрировать интересную для потенциальных инвесторов цель, создать некоррупцированные оцифрованные отношения, собрать команду для cloud-driven управления и кадры для исполнения проекта, а также определиться с территорией в киберфизическом пространстве. К сожалению, спонсоры

также достаточно легко находятся и для деструктивных проектов, ставящих целью разрушение неугодных государственных.

Что важнее для успеха на рынке - кадры или отношения? Зачастую неэффективность работы социальной системы связывают с неудачным подбором кадров на ключевых позициях, неумным выходом из чего является замена одних специалистов на других. Отношения имеют высший приоритет перед кадрами в создании эффективной компании или государственности. «Мы не ищем людей, допускающих ошибки, а исправляем структуру отношений, в которых эти ошибки стали возможными» (Стенли Гайдук).

Пример 3. Первые компании планеты достаточно просто создают эффективные филиалы в любой стране мира путем интервенции в нее системы отношений, принятой в главном офисе или на производстве. Вывод: работников самой неразвитой территории за минимальное время можно обучить технологиям, если компания будет платить существенно более высокую зарплату своим сотрудникам на фоне нищенских окладов в государственных предприятиях развивающихся стран. Выигрыш компании заключается в существенно более низком уровне зарплаты работников филиала по сравнению с сотрудниками главного офиса.

Пример 4. При равенстве социальных отношений внутри компании или развитой страны лучшие результаты действительно будут определяться наличием более квалифицированных и творческих кадров. Не последнее место в формировании рыночного успеха занимает роль лидера-идеолога. Так возникают успешные проекты: Apple, Google, Amazon, Microsoft, Alibaba в странах, где созданы толерантные метрические отношения для работы и творчества. На вопрос: что порождает коррупцию - кадры или отношения? Существует однозначный ответ: отношения, допускающие два источника коррупции: 1) чиновник-посредник, продающий 2) государственные ресурсы народу. Решение вопроса: 1) Облачное управление ресурсами без участия чиновника по заранее определенным правилам. 2) Сведение к нулю государственной собственности путем замены государственных структур (финансы, наука, образование, медицина, транспорт, юриспруденция, суд, энергетика) и предприятий на частные компании. В частных компаниях нет коррупции. Образцом цифрового киберфизического творчества, строго ориентированного на online исполнение всех желаний граждан является компания «Приватбанк» [13]. Это и есть прообраз нового типа нравственной киберфизической государственности, действи-

тельно заботящейся о каждом человеке в формате 24/7.

Что делает государство лузером на рынке: кадры или отношения? Точно не кадры, если учесть, что миллионы сограждан успешно работают по всему миру. Тогда остаются кривые отношения, которые создала политическая элита за годы независимости, включая конституционно и законодательно оформленную языковую, историческую и культурную дискриминацию граждан. Момент истории, 26 лет тому назад. Эйфория скоростной аутентичности страны затмила разум самообразованных политиков-депутатов, которые в одну ночь зачали и родили уродливую систему отношений нового государства. Казалось бы, изменить в конституции десять слов – ничего не стоит, чтобы получить страну, равную Германии. «Где есть воля, есть и путь». Но воля элиты надолго парализована безумной идеей превосходства титульной нации. Вывод: деградация государственности безвременно продолжается.

Идентичность и/или аутентичность государственности определяется не пропагандистскими пузырями мнимого превосходства одной нации (языка, истории, культуры) над другими, а вполне определенной метрикой состоятельности, которая включает: 1) Реальные доходы населения. 2) Экспорт продукции и услуг. 3) Уровень развития науки и техники, включая число нобелевских лауреатов и экспорт технологий. 4) Продолжительность жизни граждан. 5) Толерантность отношения государственной власти к языкам, истории, культуре народов. Удивительно, но факт, Республика Крым сегодня имеет три государственных языка: татарский, украинский и русский. И ни один человек в мире от такого многоязычия не страдает.

Государственность, как аппарат или сервис насилия над личностью, уходит в прошлое. Государства, где отношения насилия, ненависти и нацизма доминируют, в XXI веке исчезнут с лица земли. На смену приходят востребованные на рынке свободные киберфизические государственности с нравственными толерантными сервисами для обслуживания и развития каждого гражданина.

Базис коррупции в развивающихся странах определяется моделью, содержащей три взаимосвязанных государственных отношениями компонента: 1) народ; 2) чиновники; 3) ресурсы. Устранение всего лишь одного из них навсегда похоронит явление коррупции. Легче всего уничтожить народ, который власть считает источником коррупции. Труднее деструктурировать класс чиновников, который имеет реальную безграничную власть и ресурсы. Нельзя ликвидировать ресурсы. Самое трудное для человечества

понять, что источником коррупции являются не компоненты, упомянутые выше, а государственные отношения между ними, всегда иницилируемые и создаваемые чиновниками под себя. На рис. 8 представлены два цикла управления ресурсами: 1) <Ресурсы, чиновники, народ> представляют собой государственное управление, где посредником является армия чиновников. 2) <Ресурсы, облачный blockchain-сервис, народ> создают новую структуру взаимных связей, где из отношений исключается посредник, являющийся источником коррупции. Стоимость реализации второго цикла управления ресурсами практически равна нулю. Необходима только воля первых лиц политической элиты и поддержка народа, создаваемая массовой пропагандой нравственности отношений. Таким образом, победить коррупцию означает заменить чиновников на облачный сервис управления ресурсами, что автоматически будет означать сведение к минимуму их количества. Более сложным является путь многолетнего (50-300 лет) воспитания нравственности у чиновников, которые перестанут торговать не принадлежащими им ресурсами.

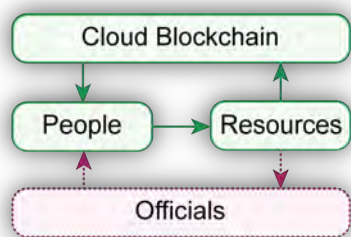


Рис. 8. Ресурсный компьютеринг: два контура управления

Модель эмоционально-логического компьютеринга оперирует несложными для понимания дискретными функциями, которые математически доказывают состоятельность не всем очевидных моделей социальных процессов и явлений. Так, практически каждая государственность имеет две ветви власти: диктатуру и демократию, которые проецируются на олигархов и народ. Для народа создаются демократические институты, которые призваны осуществлять игры для принятия судьбоносных решений по принципу большинства голосующих. Однако голосующий народ 365 дней в году зависит и управляется олигархами во власти. Поэтому народ голосует всегда "правильно" и за того, кто имеет больше денег. Нельзя выиграть в покер у партнера, который не ограничен в финансах. Следующая логическая схема, представленная на рис. 9, объясняет взаимодействие демократии и диктатуры для принятия судьбоносных государственных решений.

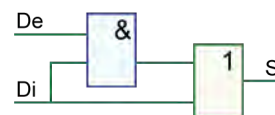


Рис. 9. Взаимодействие демократии и диктатуры. Кажется, что все справедливо, народ и капитал совместно принимают государственные решения, формально зависящие от двух переменных:

$$S = f(Di, De) = (De \wedge Di) \vee Di.$$

Однако минимизация логической функции выявляет несущественность или избыточность переменной De для формирования состояния выхода S , формирующего принятие ответственных решений для государственности:

$$S = f(Di, De) = (De \wedge Di) \vee Di = Di \wedge (De \vee 1) = Di.$$

Аналогичный результат дает взятие булевой производной по переменной De , которая равна нулю, что означает несущественность данной переменной при формировании выхода данной функции:

$$\frac{df}{d(De)} = [(1 \wedge Di) \vee Di] \oplus [(0 \wedge Di) \vee Di] = (Di \vee Di) \oplus Di = 0.$$

по второй переменной Di равна единице:

$$\begin{aligned} \frac{df}{d(Di)} &= [(De \wedge 1) \vee 1] \oplus [(De \wedge 0) \vee 0] = \\ &= [(De \wedge 1) \vee 1] \oplus [(De \wedge 0) \vee 0] = 1 \oplus 0 = 1. \end{aligned}$$

Это означает не только существенность переменной Di для формирования решений, но и полное отсутствие каких-либо других условий-ограничений, влияющих на их принятие.

Практический интерес представляет структура цифрового социального компьютеринга (рис. 10), направленная на решение задач поиска оптимального покрытия [14].

В частности, данная схема может быть использована для определения минимального числа компетентных исполнителей (руководителей) при выполнении социального проекта. Например, имеется метрика управления университетом, которая включает восемь параметров: 1) Юридическая культура для создания нравственных отношений в коллективе. 2) Управление наукой и образованием в цифровом формате. 3) Управление кадрами на основе метрического измерения социальной значимости каждого сотрудника. 4) Планирование развития университета с учетом глобальных тенденций в науке и образовании. 5) Управление и развитие киберфизической инфраструктуры университета. 6) Управление и планирование финансовой деятельности в целях привлечения государственных и частных инвестиций. 7) Международное сотрудничество, основанное на тесных киберфизических контактах университетских ученых с ведущими компаниями и

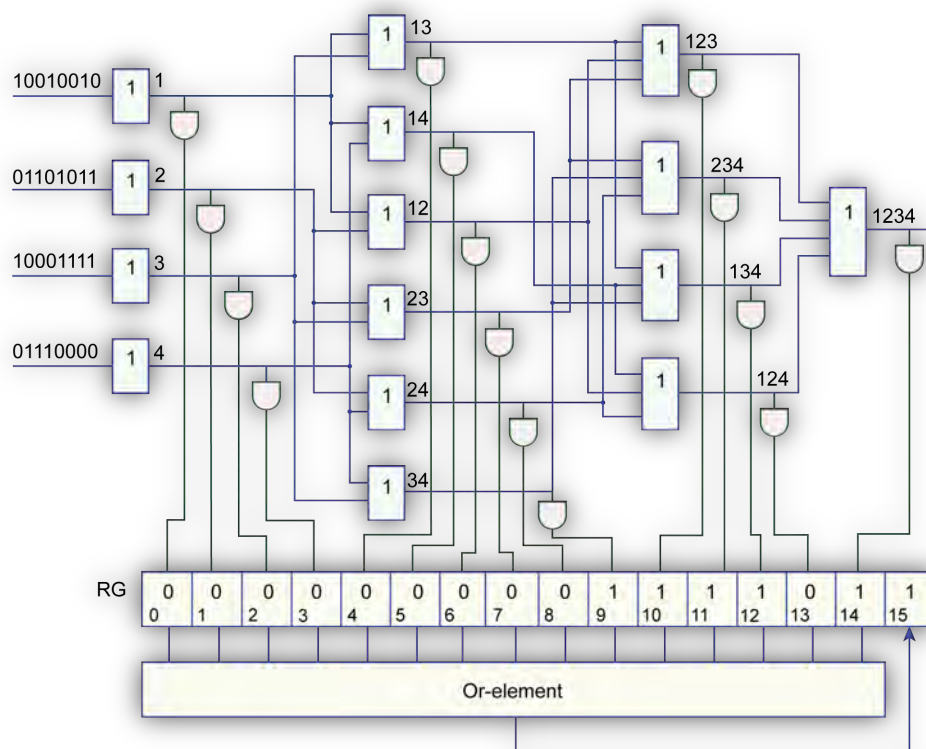


Рис. 10. Processor для определения оптимального покрытия

университетами планеты для выполнения совместных проектов в области науки и образования. 8) Воспитание физической и эстетической культуры студентов и сотрудников, основанное на индивидуальном подходе к раскрытию талантов. Следующая таблица компетенций представляет собой возможное покрытие восьми параметров метрики управления университетом четырьмя кандидатами-экспертами:

C \ P	P ₁	P ₂	P ₃	P ₄	P ₅	P ₆	P ₇	P ₈
C ₁	1	0	0	1	0	0	1	0
C ₂	0	1	1	0	1	0	1	1
C ₃	1	0	0	0	1	1	1	1
C ₄	0	1	1	1	0	0	0	0

Решение проблемы получения минимального числа экспертов, способных управлять университетом, возможно путем использования схемной структуры (см. рис. 10), позволяющей найти оптимальный результат в виде двух кандидатов C₃ и C₄, которые своими компетенциями покрывают все параметры метрического управления университетом.

5. Метрика киберфизических и социальных процессов и явлений

Социальный мир создается и определяется отношениями между людьми, которые могут быть нравственными или направленными на уничтожение человека, социальных групп. Далее рассматриваются отношения между одним, двумя,

тремя, ..., n социальными субъектами в обществе и/или государстве, которые формируют сигнатуру компьютерной модели социума. Отношения подчиняются универсальной метрике, которая устанавливает конволюционное циклическое замыкание расстояний

$$D = \sum_{i=1}^n d_i = 0$$

между ненулевым конечным числом замкнутых компонентов: $i = 1, 2, 3, \dots, n$. На рис. 11 представлена графовая интерпретация метрики для измерения отношений в киберфизическом (социальном) мире, которая основана на циклическом замыкании расстояний D между конечным числом объектов.

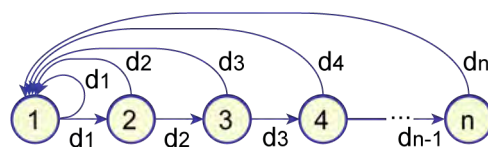


Рис. 11. Метрика измерения отношений

При этом расстояние между двумя компонентами в евклидовом метрическом пространстве определяется разностью соответствующих координат, создающих образы конечной и начальной точек геометрического вектора:

$$d(a, b) = b - a = (b_i - a_i)_{i=1}^n;$$

b =	1	2	3	1	0	5	3	1
a =	0	1	2	3	4	5	2	1
d =	1	1	1	-2	-4	0	1	0

В двоичном булевом киберпространстве различие между двумя компонентами определяется хог-суммой кодов исходных объектов:

$$d(a, b) = a \oplus b = a_i \oplus b_i;$$

a =	1	0	0	1	0	1	1	1
b =	0	1	1	1	0	1	1	0
d =	1	1	1	0	0	0	0	1

В многозначном булевом киберпространстве различие между двумя компонентами определяется симметрической разностью кодов исходных объектов, заданных символами замкнутого алфавита:

$$d(a, b) = a \Delta b = a_i \Delta b_i; A = \{0, 1, X = \{0, 1\}, \emptyset\};$$

a =	1	X	0	X	0	1	X	1
b =	0	X	1	0	X	1	1	X
d = aΔb	X	∅	X	X	X	∅	0	0

Δ	0	1	X	∅
0	∅	X	1	0
1	X	∅	0	1
X	1	0	∅	X
∅	0	1	X	∅

С учетом формул вычисления расстояний (отношений) между киберфизическими (социальными) процессами и явлениями далее предлагается реализация универсальной метрики циклического замыкания в метрическое евклидово (Е), двоичное булево (В) и многозначное (М) киберпространство. При этом D-метрика нулевой суммы расстояний циклического конволюционного замыкания легко трансформируется к принятому в научном мире формату трех аксиом: рефлексивности, симметричности и (обратной) транзитивности:

$$D(E) = \sum_{i=1}^n d_i = 0 \rightarrow \begin{cases} n=1 \rightarrow d(a, a) = 0; \\ n=2 \rightarrow d(a, b) + d(b, a) = 0; \\ n=3 \rightarrow d(a, b) + d(b, c) + d(c, a) = 0. \end{cases}$$

$$D(B) = \bigoplus_{i=1}^n d_i = 0 \rightarrow \begin{cases} n=1 \rightarrow d(a, a) = 0; \\ n=2 \rightarrow d(a, b) \oplus d(b, a) = 0; \\ n=3 \rightarrow d(a, b) \oplus d(b, c) \oplus d(c, a) = 0. \end{cases}$$

$$D(M) = \bigtriangleup_{i=1}^n d_i = \emptyset \rightarrow \begin{cases} n=1 \rightarrow d(a, a) = \emptyset; \\ n=2 \rightarrow d(a, b) \Delta d(b, a) = \emptyset; \\ n=3 \rightarrow d(a, b) \Delta d(b, c) \Delta d(c, a) = \emptyset. \end{cases}$$

Отличительное преимущество D-метрики заключается в единственности универсальной формулы для вычисления расстояний между процессами и явлениями в упомянутых пространствах, а также в возможности точного определения дуги обратного транзитивного замыкания по двум известным расстояниям, дополняющим замыкание до цикла:

$$D(E) \rightarrow d(c, a) = -[d(a, b) + d(b, c)];$$

$$D(B) \rightarrow d(c, a) = d(a, b) \oplus d(b, c);$$

$$D(M) \rightarrow d(c, a) = d(a, b) \Delta d(b, c).$$

На рис. 12 показаны примеры треугольных отношений в трех видах пространств: Евклидовом, Двоичном и Многозначном соответственно, которые представлены координатами компоненто-вершин для последующего вычисления сторон-расстояний в каждом треугольнике.

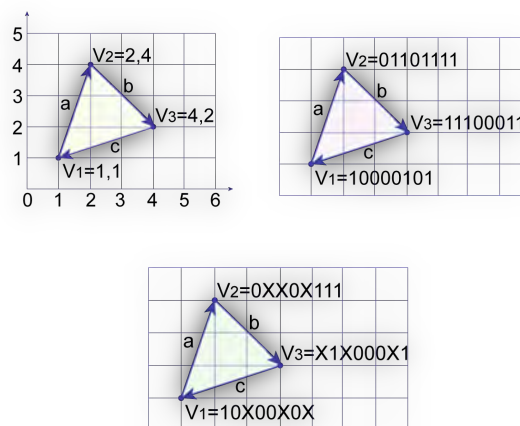


Рис.12. Треугольники в трех пространствах
Следующие примеры-таблицы демонстрируют свойства D-метрики для конволюции в 0-точку замкнутого треугольными отношениями пространства компоненто-вершин (см. рис. 12), а также для вычисления третьей стороны в треугольнике, заданном двумя другими сторонами.

V ₁	1	0	0	0	1	0	1
V ₂	0	1	1	0	1	1	1
V ₃	1	1	1	0	0	1	1
a = V ₁ ⊕ V ₂	1	1	1	0	1	0	1
b = V ₂ ⊕ V ₃	1	0	0	0	1	1	0
c = V ₃ ⊕ V ₁	0	1	1	0	0	1	0
D = a ⊕ b ⊕ c	0	0	0	0	0	0	0
c = a ⊕ b	0	1	1	0	0	1	0

V ₁	1	0	X	0	0	X	0	X
V ₂	0	X	X	0	X	1	1	1
V ₃	X	1	X	0	0	0	X	1
a = V ₁ Δ V ₂	X	1	∅	∅	1	0	X	0
b = V ₂ Δ V ₃	1	0	∅	∅	1	X	0	∅
c = V ₃ Δ V ₁	0	X	∅	∅	∅	1	1	0
D = a Δ b Δ c	∅	∅	∅	∅	∅	∅	∅	∅
c = a Δ b	0	X	∅	∅	∅	1	1	0

V ₁	1	1
V ₂	2	4
V ₃	4	2
a = V ₁ - V ₂	-1	-3
b = V ₂ - V ₃	-2	2
c = V ₃ - V ₁	3	1
D = a + b + c	0	0
c = -(a + b)	3	1

Таким образом, представленная D-метрика оперирует единственной формулой циклического конволюционного замыкания конечного числа процессов и/или явлений, из которой, в частности, следуют три известные аксиомы: рефлексивности, симметричности и обратной транзитивности.

6. Заключение

Проведенное исследование позволяет сделать следующие выводы:

1. Предложены типы эмоционально-логического компьютеринга, метрика измерения эмоциональных отношений по горизонтальным и вертикальным связям, а также правила поведения среднестатистического человека, ориентированные на создание эмоциональной логики для моделирования поведения народных масс и властных гос-

ударственных структур, а также предсказания их возможных решений.

2. Представлена киберфизическая модель государственности, которая предназначена для метрического управления ресурсами и гражданами на основе цифрового мониторинга и оценивания потребностей социальных групп в целях обеспечения качества их жизни и процветания страны путем производства и экспорта товаров и услуг. Показана структура компонентов государственного компьютеринга, упорядоченных по степени их влияния на рыночный успех: 1) Отношения. 2) Цели. 3) Кадры. 4) Управление. 5) Инфраструктура. 6) Ресурсы.

3. Сформулирована D-метрика измерения расстояний между процессами и явлениями в киберфизическом пространстве в виде универсальной формулы для точного определения дуги обратного транзитивного замыкания по двум известным расстояниям, дополняющим замыкание до конволюционного цикла.

Литература:

1. <https://www.forbes.com/sites/louiscolombus/2017/08/15/gartners-hype-cycle-for-emerging-technologies-2017-adds-5g-and-deep-learning-for-first-time/#646a4cf34be2>
2. <http://www.gartner.com/newsroom/id/3784363>
3. <http://www.wired.co.uk/article/ai-neuromorphic-chips-brains>
4. A. Gupta and R. K. Jha, "A Survey of 5G Network: Architecture and Emerging Technologies," in IEEE Access, vol. 3, pp. 1206-1232, 2015.
5. C. Zhu, V. C. M. Leung, L. Shu and E. C. H. Ngai, "Green Internet of Things for Smart World," in IEEE Access, vol. 3, pp. 2151-2162, 2015.
6. K. Christidis and M. Devetsikiotis, "Blockchains and Smart Contracts for the Internet of Things," in IEEE Access, vol. 4, pp. 2292-2303, 2016.
7. Blockchains: How They Work and Why They'll Change the World IEEE Spectrum. October 2017. <https://spectrum.ieee.org/computing/networks/blockchain-s-how-they-work-and-why-theyll-change-the-world>
8. V. Kharchenko, Y. Kondratenko, J. Kacprzyk (Eds.), "Green IT Engineering: Concepts, Models, Complex Systems Architectures", In the book series "Studies in Systems, Decision and Control" (SSDC), vol. 1, Berlin, Heidelberg: Springer International Publishing, 2017.
9. V. Kharchenko, Y. Kondratenko, J. Kacprzyk (Eds.), "Green IT Engineering: Components, Networks and Systems Implementation". In the book series "Studies in Systems, Decision and Control" (SSDC), vol. 2, Berlin, Heidelberg: Springer International Publishing, 2017.
- 10A. Zarella, N. Bui, A. Castellani, L. Vangelista and M. Zorzi, "Internet of Things for Smart Cities," in IEEE IoT Journal, vol. 1, no. 1, pp. 22-32, Feb. 2014.
- 11 https://www.gartner.com/doc/3471559?srcId=1-7578984202&utm_campaign=RM_GB_2017_TRENDS_QC_E2_What&utm_medium=email&utm_source=Eloqua&cm_mmc=Eloqua_-Email_-LM_RM_GB_2017_TRENDS_QC_E2_What_-0000
12. <http://www.gartner.com/smarterwithgartner/three-digital-marketing-habits-to-break-2/>
13. Patrick T. Hester, Kevin Adams. Systemic Decision Making. Fundamentals for Addressing Problems and Messes. Springer Nature. 412 p.

14. Hahanov Vladimir, Barkalov Alexander, Adamski Marian. Infrastructure intellectual property for SoC simulation and diagnosis service. Springer, Germany. 2011.– P. 289-330.

15. <http://dpchas.com.ua/politika/dubilet-zadumal-revoluciyu>

16. Хаханов В.И., Мищенко А.С., Емельянов И.В., Любарский М.М., Соклакова Т.И., Абдулаев В.Г. Gartner 2017 топ технологии: их анализ и применение // Paradigmata poznání. 2017. №4. P. 33-62.

Поступила в редколлегию 11.12.2017

Рецензент: д-р техн. наук, проф. Меликян В.Г.

Хаханов Владимир Иванович, д-р техн. наук, проф., главный научный сотрудник кафедры АПВТ ХНУРЭ. Научные интересы: проектирование и тестирование цифровых систем. Хобби: футбол, горные лыжи. Адрес: Украина, 61166, Харьков, пр. Науки, 14, e-mail: hahanov@icloud.com.

Мищенко Александр Сергеевич, научный сотрудник кафедры АПВТ ХНУРЭ. Научные интересы: компьютерная инженерия. Адрес: Украина, 61166, Харьков, пр. Науки, 14.

Соклакова Татьяна Игоревна, инженер кафедры АПВТ ХНУРЭ. Научные интересы: компьютерная инженерия. Адрес: Украина, 61166, Харьков, пр. Науки, 14.

Чумаченко Светлана Викторовна, д-р техн. наук, профессор, зав. кафедрой АПВТ ХНУРЭ. Научные интересы: математическое моделирование вычислительных процессов, теория рядов, методы дискретной оптимизации, инновационные формы обучения. Адрес: Украина, 61166, Харьков, пр. Науки, 14, тел. + 3805770-21-326, e-mail: svetlana.chumachenko@nure.ua

Литвинова Евгения Ивановна, д-р техн. наук, проф. кафедры АПВТ ХНУРЭ. Научные интересы: проектирование и тестирование цифровых систем. Хобби: музыка. Адрес: Украина, 61166, Харьков, пр. Науки, 14, тел. + 3805770-21-326, e-mail: litvinova_eugenia@icloud.com.

Hahanov Vladimir Ivanovich, Dr. of Tech. Sc., Prof., Chief Scientific Officer, Design Automation Department, NURE. Scientific interests: design and testing of digital systems. Hobby: football, downhill skiing. Address: Ukraine, 61166, Kharkov, Science, 14, e-mail: hahanov@icloud.com.

Mishchenko Alexandr Sergeevich, scientific researcher, Scientific interests: computer engineering. Address: Ukraine, 61166, Kharkov, Nauki Ave, 14.

Soklakova Tetiana Igorevna, Engineer, Design Automation Department. Scientific interests: computer engineering. Address: Ukraine, 61166, Kharkov, Nauki Ave, 14.

Chumachenko Svetlana Victorovna, Dr. of Tech. Sc, Professor, Head of Design Automation Department, NURE. Scientific interests: mathematical modeling of computational processes, theory of series, methods of discrete optimization, educational innovations. Address: Ukraine, 61166, Kharkov, Nauki Ave, 14, tel. + 3805770-21-326, e-mail: svetlana.chumachenko@nure.ua

Litvinova Evgenia Ivanovna, Dr. of Tech. Sc, prof., Design Automation Department, NURE. Scientific interests: design and testing of digital systems. Hobbies: music. Address: Ukraine, 61166, Kharkov, Nauki Ave, 14, e-mail: litvinova_eugenia@icloud.com.