

Н. Г. Кучук

Національний технічний університет «Харківський політехнічний інститут», Харків, Україна

СИНТЕЗ МЕРЕЖЕВОЇ МОДЕЛІ КОМП'ЮТЕРНОЇ СИСТЕМИ НА ГІПЕРКОНВЕРГЕНТНІЙ ПЛАТФОРМІ

Анотація. Метою статті є розробка методу синтезу мережевих моделей на основі часових мереж Петрі. Розроблена модель дозволить побудувати модель, що дозволяє досягти ступеня адекватності для прогнозування продуктивності програмного комплексу з необхідною достовірністю. **Результати дослідження.** Запропоновано метод синтезу часової мережі Петрі, що базується на трасуванні даних. Даний метод був використаний при моделюванні процесу функціонування комплексу програм. Основним недоліком запропонованого підходу є необхідність постійного збору виміральної інформації в комп'ютерній системі. Показано, що від точності виміральної інформації залежить ступінь адекватності моделі. Проаналізована адекватність опису динаміки досліджуваного процесу. Для оцінки ефективності запропонованої моделі вона була використана для прогнозу продуктивності пакета композитних застосунків, що використовуються при продажі авіаквитків. **Висновок.** Для практичних застосувань у більшості випадків достатньо синтезувати мережеву модель комп'ютерної системи на гіперконвергентній платформі на основі 30-35 вимірювань. Вибір більшого значення з метою зменшення розміру довірчого інтервалу одержуваних оцінок не буде виправданим, якщо основне завдання прогнозування - мінімізація помилки прогнозу.

Ключові слова: гіперконвергентна платформа; модель, мережа Петрі.

Вступ

Постановка завдання. Конвергентні та гіперконвергентні рішення для базових комп'ютерних мереж за рахунок централізації управління суттєво зменшують витрати на обслуговування [1]. При такому технологічному рішенні передбачається об'єднання пам'яті, обчислювальних, програмних і мережевих ресурсів в пул, заздалегідь об'єднаних для роботи в дата-центрі [2]. Управління відбувається через загальну консоль адміністрування [3]. Але на сьогодні недостатньо досліджені питання аналізу якості комплексів програм, що функціонують в гіперконвергентних середовищах. Особливо важливим для таких систем є аналіз продуктивності. Для багатьох комп'ютерних систем необхідно забезпечення необхідного рівня цього показника. На етапі розробки Software це є важкою практичним завданням. Навіть налагоджені відтестовані на коректність програми в реальних умовах можуть показувати незадовільну продуктивність. Це відбувається внаслідок того, що при складанні та виконанні тестів необхідно враховувати паралелізм обчислювальних процесів. Також час надходження і обробки вхідних даних має бути додатковим параметром кожного тесту. Це вимагає неприйнятно великих обсягів тестування і значно ускладнює інтерпретацію його результатів. Одним із шляхів вирішення зазначеного вище завдання є використання моделей програмних систем. Модель дозволяє врахувати специфіку організації паралельних обчислювальних процесів в гіперконвергентних системах. Крім того, модель може врахувати особливості реалізації Software.

Аналіз літератури. Загальним підходом до аналізу складної системи є оцінка поведінки її моделі під впливом подій, що генеруються моделлю зовнішніх впливів [4-7]. Для комп'ютерних систем множина вхідних завдань називається робочим навантаженням [8]. Залежно від цілей дослідження під робочим навантаженням можуть розумітися як вхід-

ні дані, так і програми [9]. Для всіх відомих методів моделювання програмних систем досить складним є опис моделі робочого навантаження [10-14]. При цьому необхідно вирішувати окреме завдання, пов'язану з оцінкою їх точності та адекватності [15]. Широко поширені графові моделі програм будуються на основі вивчення статичних структур їх початкового тексту [16, 17]. Однак вони не враховують реальної динаміки поведінки процесів, а відображають погляд дослідника або розробника на передбачувану поведінку системи [18]. Найбільш відомі трасові моделі, засновані на даних вимірювань реальних системи, як вказувалося раніше, надмірно об'ємні і недостатньо гнучкі [19, 20]. Однак всі розглянуті підходи не в повній мірі враховують особливості проектування та реалізації сучасних програмних платформ, зокрема гіперконвергентних. Внаслідок цього недостатньо дослідженими залишилися питання, пов'язані з аналізом якості комплексів програм, що функціонують в гіперконвергентних середовищах [21, 22]. Особливо це позначається при розрахунку такого показника, як продуктивність.

Мета статті. Важливим завданням при розробці програмних комплексів є прогнозування продуктивності. Метою статті є розробка методу синтезу мережевих моделей на основі часових мереж Петрі. Розроблена модель дозволить побудувати модель, що дозволяє досягти ступеня адекватності для прогнозування продуктивності програмного комплексу з необхідною достовірністю.

1. Синтез часової мережі Петрі, що базується на трасуванні даних

Для побудови моделі використовується концептуальна модель функціонування програмної системи. Метод заснований на поданні трас подій, що відбуваються в системі, часовими мережами Петрі. Для визначення часу кожного переходу використовуються випадкові величини. Кожна випадкова величина описується законом розподілу загального

вигляду. Введемо поняття ансамблю переходів. Він описується повною групою несумісних подій. Кожен перехід в складі ансамблю визначається ймовірністю відповідної події. Реалізація ансамблю переходів дозволяє організувати вибір напрямку розвитку процесу. Пропонується метод синтезу часової мережі Петрі з використанням трасувань даних. Метод використовує ансамблі переходів і складається з таких кроків.

Крок А. Виділення N типів часових подій. Ці події істотні з точки зору цілей моделювання: Вони складають множину Ω :

$$\Omega = \{\omega_1, \omega_2, \dots, \omega_N\}, \quad (1)$$

де ω_i – i -й тип часової події.

Для побудови моделі робочого навантаження мережі в якості подій розглядаються надходження заявок на обробку. Для моделювання комплексу програм розглядаються процеси обробки заявок конкретних типів за заданими алгоритмами.

Крок В. Організація збору інформації про параметри подій з Ω ; виділення трас подій.

Крок С. Виділення станів програмної системи. Множина станів S формується на основі аналізу отриманих трас:

$$S = \{s_1, s_2, \dots, s_M\}, \quad s_j = (\omega_{j1}, \omega_{j2}, \dots, \omega_{jk}), \quad (2)$$

де j_k – кількість часових подій, що визначають вектор s_j ; s_j – вектор опису події j .

Крок D. Опис траси спрямованим зваженим графом. Множина вершин даного графа T - виділені стани, дуги графа відображають послідовності переходів від одного стану до іншого. Вага кожної дуги графа визначається як ймовірність відповідного переходу.

Крок E. Побудова часової мережі Петрі. Мережа NP будується на основі графа подій таким чином:

$$NP = \{P, \Omega_{tr}, F, H, G_V, G_S, M_0\}, \quad (3)$$

де $P = \{p_1, p_2, \dots, p_L\}$ – множина позицій умов

$$Track = (\omega_2, \omega_3, \omega_1, \omega_2, \omega_3, \omega_4, \omega_2, \omega_1, \omega_5, \omega_2, \omega_3, \omega_4), \text{card}(Track) = 12. \quad (6)$$

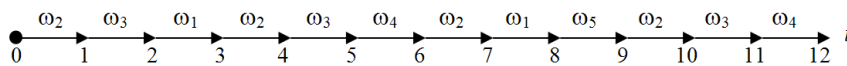


Рис. 1. Подіє-часова траса

Послідовність подій (6) є марківським ланцюгом. У цьому випадку множина S станів системи ізоморфна множині Ω . Граф, відповідний трасі (6), наведено на рис. 2. Відзначимо, що дуги графа на рис. 2, які виходять із вершин s_4 і s_5 , реалізуються звичайними часовими переходами. Дуги графа, які виходять із вершин s_1, s_2 і s_3 , реалізуються ансамблями переходів. Кількість переходів в ансамблі визначається кількістю вихідних дуг. Імовірність спрацювання переходу з ансамблю визначається вагою відповідної дуги.

виконання подій; вона визначається наявністю вхідних дуг графу подій; L – кількість вхідних дуг;

Ω_{tr} – множина переходів, що відповідають множині подій Ω ;

$$P \cap \Omega_{tr} = \emptyset; P \cup \Omega_{tr} \neq \emptyset; \quad (4)$$

$F : P \times \Omega_{tr} \rightarrow \{0, 1\}$ – булева функція передування множини позицій P і переходів Ω_{tr} ; функція визначається вхідними дугами графа подій;

$H : \Omega_{tr} \times P \rightarrow \{0, 1\}$ – булева функція слідування множини переходів Ω_{tr} і позицій P ; функція визначається вихідними дугами графа подій;

$G_V : \Omega_{tr} \times R_V \rightarrow f(Z_V)$ – функція відповідності між множинами переходів Ω_{tr} і множиною випадкових величин часу виконання подій R_V ; елементи R_V розподіляються за законом розподілу Z_V ; закон розподілу і його параметри визначаються на основі даних траси;

$G_S : \Omega_{tr} \times P_S \rightarrow \{0, 1\}$ – функція відповідності між множинами переходів Ω_{tr} і множиною ймовірностей їх спрацювання P_S ; елементи множини P_S дорівнюють вагам графа подій;

$M_0 : P \rightarrow \{0, 1, 2, \dots\}$ – початковий розподіл маркерів часової мережі Петрі; в одній вершині мережі може бути більше одного маркера.

2. Застосування синтезу часової мережі Петрі при моделюванні комплексу програм

Розглянемо приклад застосування запропонованого підходу для моделювання найпростішої програмної системи. Нехай пакет композитних застосунків використовує п'ять різних запитів до сховища даних. Запити утворюють множину

$$\Omega = \{\omega_1, \omega_2, \omega_3, \omega_4, \omega_5\}, N = 5. \quad (5)$$

В результаті експериментального виконання пакету була отримана така траса (рис. 1):

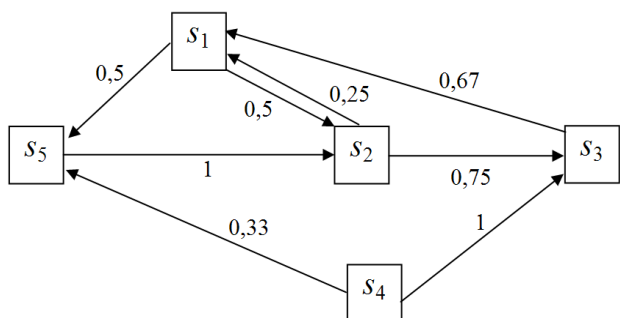


Рис. 2. Граф траси рис. 1

Відповідно до кроків А - Е наведеного вище алгоритму, побудуємо часову мережу Петрі (формула (3) та рис. 3).

Початковий маркер потрапляє в вершину p_2 , тому що подія ω_2 означає початок обробки пакета композитних застосунків.

Додатково введена вершина p_0 . Вона відповідає умові «запуск траси». Перехід ω_0 генерує випадкове час запуску пакета.

Проведемо аналіз запропонованого методу синтезу часової мережі Петрі з використанням трасувань даних для моделювання комплексу програм.

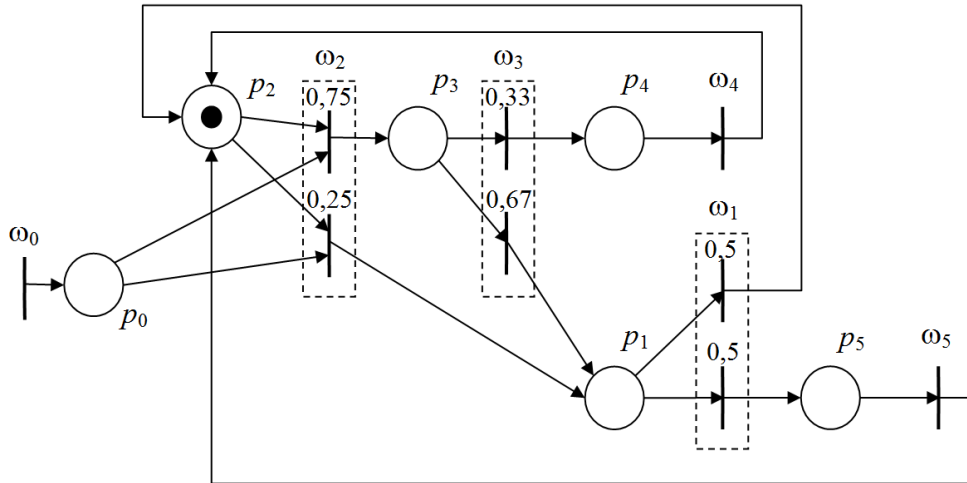


Рис. 3. Часова мережа Петрі траси на рис. 1

Необхідні умови для застосування методу:

а) наявність інформації про послідовність і тривалості мережевих подій;

б) застосування вимірювальних моніторів;

с) припущення про стаціонарності ймовірностей переходів системи з обраним станів.

Перевагами запропонованого методу є:

а) орієнтованість на дослідження паралельних процесів;

б) можливість використання різних рівнів деталізації і ієрархічного моделювання;

с) можливість дослідження за допомогою однієї моделі як робочого навантаження, так і комплексу програм;

д) можливість декомпозиції і складання моделей;

е) простота врахування станів зовнішнього середовища.

Основний недолік методу - необхідність постійного збору вимірювальної інформації в комп'ютерній системі. Від точності вимірювальної інформації залежить ступінь адекватності моделі.

3. Оцінка адекватності мережевої моделі

Основна вимога до моделі – адекватність реальної системі. Процеси функціонування реальних програмних комплексів неможливо описати повно і детально. Це, в першу чергу, обумовлено їх значною складністю. Тому можна говорити лише про ступінь адекватності моделі. Підвищення ступеня адекватності може бути досягнуто за рахунок використання різних рівнів деталізації. Для оцінки ступеня адекватності моделі необхідно вибрати критерій, відповідний розв'язуваній задачі. В даній статті оцінка адекватності заснована на ступені відповідності поведінки моделі системи. Іншими словами аналізується адекватність опису динаміки досліджуваного процесу.

На кроці С запропонованого вище алгоритму визначено настроюваний параметр M – кількість станів системи. Цей параметр залежить від тривалості передісторії D . При $D=0$ маємо марківський процес без передісторії. Використання гіпотези про марковість досліджуваних процесів ($D=0$) призводить до отримання найбільш компактною стохастичної моделі. Збільшення D аж до розгляду довжини історії всієї траси, дає більш точний результат. Однак при цьому істотно зростає складність трасування моделі.

Верхня межа можливого числа станів системи розраховується як

$$M_{max} = C_N^D \cdot D! = N! / (N - D)! \quad (7)$$

При моделюванні комплексу програм необхідно вирішити задачу вибору конкретного значення тривалості врахування передісторії розвитку процесів. При цьому треба оцінити достовірність подання поведінки системи трасовою мережевою моделлю.

Задамо множину можливих початкових станів системи таким чином:

$$S_0 = \{s_1^{(0)}, s_2^{(0)}, \dots, s_N^{(0)}\} \quad (8)$$

Відмітимо, що S_0 є ізоморфним Ω , причому $D=0$.

Ітеративно збільшуючи значення D можна добитися необхідного рівня адекватності моделі. При цьому зростає розмірність множини S_0 . Тому необхідно не перевищити допустимий рівень складності моделі.

Нехай на кроці ξ ітераційного процесу розглядається множина подій S_ξ . Для даної множини визначимо ймовірності переходу системи зі стану i в

стан j за n кроків ($n \in \mathbb{N}$ – натуральне число) – $P_{ij}^{(n,\xi)}$. Розглянемо матрицю перехідних ймовірностей марківського ланцюга – $Q^{(\xi)} = \left\| q_{ij}^{(\xi)} \right\|$, яка має розмірність ρ . Тоді $P_{ij}^{(0,\xi)} = q_{ij}^{(\xi)}$.

Введемо виробляючу функцію процесу

$$\tilde{P}_{ij}^{(\xi)}(\zeta) = \sum_{n=0}^{\infty} P_{ij}^{(n,\xi)} \zeta^n, \quad |\zeta| < 1. \quad (9)$$

Пмноживши обидві частини рівності (9) на $\zeta \cdot q_{ij}^{(\xi)}$ і підсумовуючи по $i = \overline{1, \rho}$, отримуємо співвідношення

$$\begin{aligned} s \sum_{i=1}^{\rho} q_{ij}^{(\xi)} \tilde{P}_{ij}^{(\xi)}(\zeta) &= \\ &= \sum_{n=0}^{\infty} \sum_{i=1}^{\rho} q_{ki} P_{ij}^{(n,\xi)} \zeta^{n+1} = \tilde{P}_{ij}^{(\xi)}(\zeta) - P_{kj}^{(0)}, \end{aligned} \quad (10)$$

яке визначає систему рівнянь вигляду

$$\tilde{P}_{kj}^{(\xi)}(\zeta) - \zeta \sum_{i=1}^{\rho} q_{ij}^{(\xi)} \tilde{P}_{ij}^{(\xi)}(\zeta) = P_{kj}^{(0)}, \quad (11)$$

$$k = \overline{1, \rho}, \quad j = \overline{1, \rho}.$$

Їх рішеннями при фіксованих k і ζ є функції вигляду

$$\tilde{P}_{ij}^{(\xi)}(\zeta) = \frac{G_{ij}(\zeta)}{D(\zeta)}. \quad (12)$$

Дріб (12) розкладається на прості дробі

$$\tilde{P}_{ij}^{(\xi)}(\zeta) = \sum_{\lambda=1}^{\rho} \frac{g_{ij}^{(\lambda)}}{1 - \zeta \cdot r_{\lambda}}, \quad (13)$$

де r_{λ} – власні значення матриці $Q^{(\xi)}$ перехідних ймовірностей; $g_{ij}^{(\lambda)}$ – коефіцієнти, отримані при розкладанні.

Нехай h – довільний лівий власний вектор матриці $Q^{(\xi)}$, тобто

$$h Q^{(\xi)} = r_{\lambda} h. \quad (14)$$

Розглянемо суму елементів векторів лівої і правої частини рівності (14).

$$\sum_{j=1}^{\rho} \sum_{i=1}^{\rho} h_i q_{ij}^{(\xi)} = \sum_{j=1}^{\rho} h_i \sum_{i=1}^{\rho} q_{ij}^{(\xi)} = \sum_{j=1}^{\rho} h_i = r_{\lambda} \sum_{j=1}^{\rho} h_i. \quad (15)$$

За умови, що всі h_i є позитивними, $r_{\lambda} = 1$ означає спектральний радіус матриці $Q^{(\xi)}$. Отже, все її власні значення задовольняють нерівності

$$|r_{\lambda}| \leq 1, \quad (16)$$

тому

$$\frac{1}{1 - \zeta \cdot r_{\lambda}} = \sum_{i=0}^{\infty} \zeta^i r_{\lambda}^i, \quad (17)$$

а з (13) слідує, що

$$P_{ij}^{(n,\xi)} = \sum_{\lambda=1}^{\rho} g_{ij}^{(\lambda)} r_{\lambda}^n. \quad (18)$$

Для знаходження значень коефіцієнтів $g_{ij}^{(\lambda)}$ скористаємося ітераційністю процесу:

$$\begin{aligned} P_{ij}^{(n+1,\xi)} &= \sum_{\lambda=1}^{\rho} g_{ij}^{(\lambda)} r_{\lambda}^{n+1} = \sum_{k=1}^{\rho} q_{ik}^{(\xi)} P_{kj}^{(n,\xi)} = \\ &= \sum_{k=1}^{\rho} q_{ik}^{(\xi)} \sum_{\lambda=1}^{\rho} g_{kj}^{(\lambda)} r_{\lambda}^n = \sum_{\lambda=1}^{\rho} r_{\lambda}^n \sum_{k=1}^{\rho} q_{ik}^{(\xi)} g_{kj}^{(\lambda)}. \end{aligned} \quad (19)$$

Із (19) отримаємо:

$$g_{ij}^{(\lambda)} r_{\lambda} = \sum_{k=1}^{\rho} q_{ik}^{(\xi)} g_{kj}^{(\lambda)}. \quad (20)$$

З іншого боку,

$$\begin{aligned} P_{ij}^{(n+1,\xi)} &= \sum_{\lambda=1}^{\rho} g_{ij}^{(\lambda)} r_{\lambda}^{n+1} = \sum_{k=1}^{\rho} P_{ik}^{(n,\xi)} q_{kj}^{(\xi)} = \\ &= \sum_{k=1}^{\rho} \sum_{\lambda=1}^{\rho} g_{ik}^{(\lambda)} r_{\lambda}^n q_{kj}^{(\xi)} = \sum_{\lambda=1}^{\rho} r_{\lambda}^n \sum_{k=1}^{\rho} g_{ik}^{(\lambda)} q_{kj}^{(\xi)}. \end{aligned} \quad (21)$$

отже,

$$g_{ij}^{(\lambda)} r_{\lambda} = \sum_{k=1}^{\rho} g_{ik}^{(\lambda)} q_{kj}^{(\xi)}. \quad (22)$$

У матричній формі вирази (20) і (22) можуть бути записані як

$$G^{(\lambda)} Q^{(\xi)} = r_{\lambda} G^{(\lambda)}, \quad (23)$$

$$Q^{(\xi)} G^{(\lambda)} = r_{\lambda} G^{(\lambda)}. \quad (24)$$

Таким чином, стовпці матриці $G^{(\lambda)}$ є правими власними векторами матриці $Q^{(\xi)}$. Вони визначаються при $r = r_{\lambda}$ ненульовими рішеннями системи рівнянь

$$\sum_{k=1}^{\rho} q_{kj}^{(\xi)} x_k - r \cdot x_i = 0; \quad i = \overline{1, \rho}. \quad (25)$$

Рядки матриці $G^{(\lambda)}$ є лівими власними векторами. Вони визначаються ненульовими рішеннями $y_j^{(\lambda)}$ системи

$$\sum_{k=1}^{\rho} y_k q_{kj}^{(\xi)} - r \cdot y_j = 0; \quad j = \overline{1, \rho}. \quad (26)$$

Тоді з точністю до постійного множника $C^{(\lambda)}$ визначаємо

$$g_{ij}^{(\lambda)} = C^{(\lambda)} x_i^{(\lambda)} y_j^{(\lambda)}. \quad (27)$$

Значення $C^{(\lambda)}$ розраховуються виходячи з ортонормованості лівих і правих власних векторів:

$$C^{(\lambda)} \cdot \sum_{k=1}^{\rho} x_k^{(\lambda)} y_k^{(\lambda)} = 1 \Rightarrow$$

$$\Rightarrow C^{(\lambda)} = \left(\sum_{k=1}^p x_k^{(\lambda)} y_k^{(\lambda)} \right)^{-1} \quad (28)$$

Таким чином можна розрахувати теоретичні значення перехідних ймовірностей $P_{ij}^{(n,\xi)}$.

За даними спостережимої траси станів реальної системи обчислюються частоти переходу системи зі стану i до стану j за $k = \overline{1, N}$ кроків. Вони представляються матрицею $V_i = \{v_{ij}^{(k)}\}$, при цьому

$$\sum_{(j)} v_{ij}^{(k)} = n_i^{(k)} \quad (29)$$

Припустимо, що кількості спостережень $n_i^{(k)}$ досить для застосування критерію Пірсона. Тоді в якості статистики, що характеризує відхилення експериментальних частот від відповідних теоретичних значень, можна прийняти величину

$$\begin{aligned} \chi_n^2(V_i) &= \sum_{(j)} \frac{[v_{ij}^{(k)} - n_i^{(k)} \cdot P_{ij}^{(k)}]^2}{n_i^{(k)} \cdot P_{ij}^{(k)}} = \\ &= \sum_{(j)} \frac{[v_{ij}^{(k)}]^2}{n_i^{(k)} \cdot P_{ij}^{(k)}} - n_i^{(k)}. \end{aligned} \quad (30)$$

Задамося рівнем значущості α . Тоді гіпотеза про поведінку системи з кількістю станів L відкидається при перевищенні величини $\chi_n^2(V_i)$. Дана величина розподілена за законом χ^2 з $L-1$ ступенями свободи. Вона відповідає табличному значенню $\chi_{1-\alpha, L-1}^2$.

При великих значеннях L можна використовувати інформаційний критерій

$$J_C = \frac{N_i^{(k)} - M(H_i^{(k)})}{\sqrt{D(H_i^{(k)})}} \quad (31)$$

де:

$$N_i^{(k)} = - \sum_{j=1}^L \frac{v_{ij}^{(k)}}{n_i^{(k)}} \cdot \ln \left(\frac{v_{ij}^{(k)}}{n_i^{(k)}} \right) - \quad (32)$$

статистична оцінка ентропії емпіричного розподілу;

$$M(H_i^{(k)}) = h_i^{(k)} - (L-1) / n_i^{(k)} - \quad (33)$$

математичне сподівання;

$$D(H_i^{(k)}) = \frac{1}{n_i^{(k)}} \left(\sum_{j=1}^L P_{ij}^{(k)} \cdot \ln^2(P_{ij}^{(k)}) - [h_i^{(k)}]^2 \right) - \quad (34)$$

дисперсія ентропії теоретичного розподілу;

$$h_i^{(k)} = - \sum_{j=1}^L P_{ij}^{(k)} \cdot \ln(P_{ij}^{(k)}) \quad (35)$$

Статистика J_C розподілена нормально з ну-

льовим математичним сподіванням і одиничною дисперсією. Для заданого рівня значущості α і квантиля розподілу $u_{1-\alpha}$ виконується рівність

$$|J_C| \leq u_{1-\alpha} \quad (36)$$

Інформаційний критерій за потужністю практично не поступається критерію Пірсона. У ньому ймовірність відкинути вірну гіпотезу при великій кількості спостережень значно нижче.

При великій кількості станів системи складно отримати вирази для теоретичних ймовірностей $P_{ij}^{(n)}$. У цьому випадку в якості альтернативи аналітичному підходу значення теоретичних ймовірностей можуть бути отримані методом Монте-Карло, тобто при імітаційному моделюванні поведінки відповідного марківського ланцюга. При цьому завдання перевірки гіпотези про відповідність експериментальних і модельних ймовірнісних розподілів може бути вирішена на основі критерію вигляду

$$\chi_n^2(V_i) = n_i^{(k)} m_i^{(k)} \sum_{(j)} \frac{\left(\frac{v_{ij}^{(k)}}{n_i^{(k)}} - \frac{\omega_{ij}^{(k)}}{v_{ij}^{(k)} + \omega_{ij}^{(k)}} m_i^{(k)} \right)^2}{v_{ij}^{(k)} + \omega_{ij}^{(k)}}, \quad (37)$$

де $\omega_{ij}^{(k)}$ – частоти переходу моделі із стану i до стану j за $k = \overline{1, N}$ кроків; $\sum_{(j)} \omega_{ij}^{(k)} = m_i^{(k)}$.

Таким чином, запропонований підхід дозволяє оцінити ступінь адекватності розроблених мережевих моделей на основі тимчасових мереж Петрі, врахувати передісторію подій і процесів та обґрунтувати вибір довжини траси.

4. Експеримент

Для оцінки ефективності запропонованої моделі вона була використана для прогнозу продуктивності пакета композитних застосунків, що використовуються при продажі авіаквитків.

Для побудови мережевої моделі, яка описує пакет композитних додатків, використовувалися трасувальні дані. Дані були отримані при 400 тестових прогонах розробляється підсистеми. Загальна кількість викликів запитів склало 75000.

Модель пакета задана подієвим графом. Граф описується матрицею перехідних ймовірностей. Матриця була побудована на результатах аналізу трасувань даних. На базі матриці була сформована часова мережа Петрі, що описує процес функціонування пакету композитних застосунків.

Для перевірки адекватності первинної моделі було проведено імітаційне моделювання. В результаті були отримані частотні характеристики переходу системи з обраного стану в усі інші стани за K кроків.

Для оцінки адекватності моделі використовувався критерій хі-квадрат $\chi_n^2(k)$, де $k (k \in \overline{1, K})$ – кількість кроків.

Для $K = 10$ були отримані такі результати:

$$\begin{aligned} \chi_n^2(1) &= 1.1, \chi_n^2(2) = 1.3, \chi_n^2(3) = 12.5, \chi_n^2(4) = 15.8, \\ \chi_n^2(5) &= 26.1, \chi_n^2(6) = 109.4, \chi_n^2(7) = 133.0, \\ \chi_n^2(8) &= 226.1, \chi_n^2(9) = 279.1, \chi_n^2(10) = 413.7. \end{aligned}$$

На підставі цих даних був зроблений висновок про істотне порушення адекватності при $K > 5$. Основна причина - припущення про те, що процеси переходів є марківськими. Для побудови більш адекватної моделі в даному випадку необхідно врахувати передісторію викликів. Оцінка відхилення параметрів закону розподілу часу виконання запиту при моделюванні від реального наведена в табл. 1.

Застосування критерію χ^2 дозволило отримати кількісну оцінку відповідності поведінки моделі реальної системі. Підвищення ступеня достовірності результатів моделювання підтверджується експериментальними даними, отриманими при різних

припущеннях про час перебування системи в різних станах. Проведені експерименти свідчать про можливість налагодження оптимальних параметрів розробленого методу на основі контрольної вибірки. При цьому слід враховувати залежність розміру R_b

довірчого інтервалу оцінки \hat{F} – нахила прогнозуєної прямої від значення N – кількості вимірів. Графік такої залежності при різних постійних значеннях оцінки дисперсії s^2 представлений на рис. 4.

Таблиця 1 – Оцінка математичного сподівання і дисперсії часу виконання за результатами імітаційного моделювання

Закон розподілу часу виконання запитів T	Реальна система		Модель	
	М	Д	М	Д
Детермінований, $T = 0.01$ с	1.86	2.65	1.99	3.85
Рівномірний, $T = [0, 0.01]$	0.93	0.52	1.01	0.97
Експоненційний, $M(T) = 0.01$	1.86	2.68	1.95	3.72

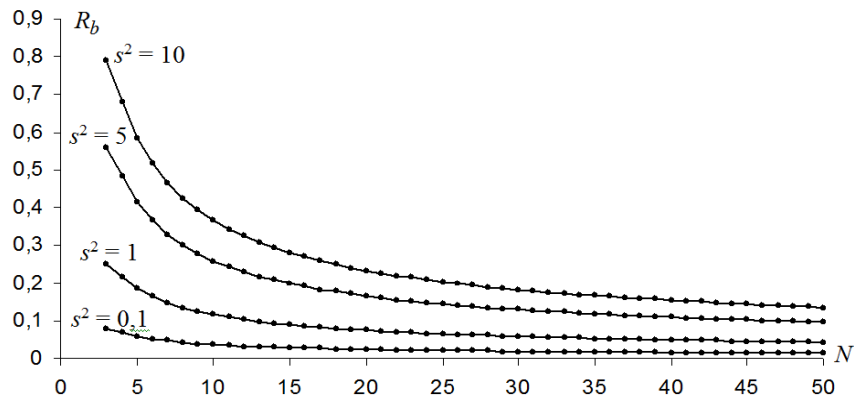


Рис. 4. Залежність розміру довірчого інтервалу від кількості вимірювань при $\alpha = 0,05$

Отже, як бачимо з графіків, в більшості практичних випадків прогнозування часових рядів може бути рекомендовано значення N , що не перевищує 30-35. Вибір більшого значення з метою зменшення розміру довірчого інтервалу одержуваних оцінок не буде виправданим, враховуючи основне завдання методу прогнозування - мінімізацію помилки прогнозу.

Висновки

Запропоновано метод синтезу часової мережі Петрі, що базується на трасуванні даних. Даний метод був використаний при моделюванні процесу функціонування комплексу програм. Основним недоліком запропонованого підходу є необхідність постійного збору вимірювальної інформації в комп'ютерній системі. Показано, що від точності вимірювальної інформації залежить ступінь адекватності моделі. Проаналізована адекватність опису динаміки досліджуваного процесу. Для оцінки ефективності запропонованої моделі вона була використана для прогнозу продуктивності пакета композитних застосунків, що використовуються при продажі авіаквитків. Для

практичних застосувань у більшості випадків достатньо синтезувати мережеву модель комп'ютерної системи на гіперконвергентній платформі на основі 30-35 вимірювань. Вибір більшого значення з метою зменшення розміру довірчого інтервалу одержуваних оцінок не буде виправданим, якщо основне завдання прогнозування - мінімізація помилки прогнозу.

Напрямок подальших досліджень: проведення порівняльного аналізу різних методів мінімізації помилки прогнозу у гіперконвергентних системах.

Вдячність

Цю роботу було частково профінансовано Європейським Союзом у контексті проекту «dComFra – Digital competence framework for Ukrainian teachers and other citizens» (Project Number: 598236-EPP-1-2018-1-LT-EPPKA2-CBHE-SP) за програмою ERASMUS+. Підтримка Європейською Комісією створення цієї роботи не означає повного схвалення її змісту, який віддзеркалює лише погляди авторів. Комісія не може нести відповідальності за будь-яке використання інформації, яку розміщено в цій роботі.

СПИСОК ЛІТЕРАТУРИ

1. White Paper: Riverbed Hyper-converged Edge, available at: <https://www.riverbed.com/document-repository/white-paper--riverbed-hyper-converged-edge.html>.

2. Черняк, Л. (2012), "Время конвергентных инфраструктур", Открытые системы. СУБД, № 4, available at: <https://www.ospr.ru/os/2012/04/13015754/>.
3. Ганьжа, Д. (2016), "Гиперконвергенция: ИТ-инфраструктура на раз, два, три", /Журнал сетевых решений, № 5, available at: www.ospr.ru/lan/2016/05/13049349.
4. Кучук Н. Г. Метод розгалуження запитів до сховищ даних систем, що мають гіперконвергентну інфраструктуру / Н. Г. Кучук, І. С. Зиков, В. І. Панченко // Системи управління, навігації та зв'язку. – Полтава : ПНТУ, 2019. – Вип. 5(57). – С. 51-54. – DOI: <https://doi.org/10.26906//SUNZ.2019.5.051>
5. Кучук Г. А. Концептуальний підхід до синтезу структури інформаційно-телекомунікаційної мережі / Г. А. Кучук, І. В. Рубан, О. П. Давікоза // Системи обробки інформації. – Х.: ХУ ПС, 2013. – Вип. 7 (114). – С. 106 – 112.
6. Кучук Г.А. Метод мінімізації середньої затримки пакетів у віртуальних з'єднаннях мережі підтримки хмарного сервісу / Г.А. Кучук, А.А. Коваленко, Н.В. Лукова-Чуйко // Системи управління, навігації та зв'язку. – Полтава: ПНТУ, 2017. – Вип. 2(42). – С. 117-120.
7. Свиридов А. С., Коваленко А. А., Кучук Г. А. Метод перерозподілу пропускної здатності критичної ділянки мережі на основі удосконалення ON/OFF-моделі трафіку. *Сучасні інформаційні системи*. 2018. Т. 2, № 2. С. 139–144. DOI: <https://doi.org/10.20998/2522-9052.2018.2.24>
8. Mozhaev O. Multiservice network security metric / O. Mozhaev, N. Kuchuk, M. Mozhaev, M. Lohvynenco // IEEE Advanced information and communication technologies-2017. Proc. of the 2th Int. Conf. – Lviv, 2017. – P. 133-136. – DOI: <https://doi.org/10.1109/AIAC.2017.8020083>
9. Кучук Г.А. Распределение каналов по трактам узла коммутации при адаптивной маршрутизации / Г.А. Кучук // Вестник НТУ «ХПИ». – Х.: НТУ «ХПИ», 2003. – № 26. – С. 167 – 172.
10. Кучук Г.А. Розрахунок навантаження мультисервісної мережі / Г.А. Кучук, Я.Ю. Стасєва, О.О. Болюбаш // Системи озброєння і військова техніка. – 2006. – № 4 (8). – С. 130 – 134.
11. Кучук Г.А. Синтез стратифікованої інформаційної структури інтеграційної компоненти гетерогенної складової Єдиної АСУ Збройними Силами України / Г. А. Кучук, О. П. Давікоза // Наука і техніка Повітряних Сил Збройних Сил України. – 2013. – № 3 (12). – С. 154-158.
12. Кучук Г.А. Управління трафіком мультисервісної розподіленої телекомунікаційної мережі / Г.А. Кучук // Системи управління, навігації та зв'язку. – К.: ЦНДІ НіУ, 2007. – Вип. 2. – С. 18-27.
13. Кучук Г.А. Минимизация загрузки каналов святы вычислительной сети / Г.А. Кучук // Системи обробки інформації. – Х.: НАНУ, ПАНМ, ХВУ, 1998. – Вип. 1(5). – С. 149-154.
14. Sivaram, M., Batri, K., Amin Salih, Mohammed and Porkodi V. (2019), "Exploiting the Local Optima in Genetic Algorithm", *Indian Journal of Science and Technology*, Vol. 12, Is. 1, doi: <http://doi.org/10.17485/ijst/2019/v12i1/139577>
15. Ruban, I. Redistribution of base stations load in mobile communication networks / I. Ruban, N. Kuchuk, A. Kovalenko // Innovative technologies and scientific solutions for industries. – 2017. – No 1 (1) – P. 75-81.
16. Кучук Г.А., Мохаммад А.С., Коваленко А.А. Метод параметрического управления передачей данных для модификации транспортных протоколов беспроводных сетей. *Системи обробки інформації*. 2011. № 8(98). С. 211-218.
17. Кучук Г. А., Можжаев А. А. Прогнозирование трафика для управления перегрузками интегрированной телекоммуникационной сети // Радиозлектронные и компьютерные системы. 2007. № 8 (27). С. 261-271.
18. Кучук, Г.А. Метод уменьшения времени передачи данных в беспроводной сети / Г.А. Кучук, А.С. Мохаммад, А.А. Коваленко // Системи управління, навігації та зв'язку. – К.: ЦНДІ НіУ, 2011. – Вип. 3 (19). – С. 209–213.
19. Кучук Г.А. Метод мінімізації середньої затримки пакетів у віртуальних з'єднаннях мережі підтримки хмарного сервісу / Г.А. Кучук, А.А. Коваленко, Н.В. Лукова-Чуйко // Системи управління, навігації та зв'язку. – Полтава . ПНТУ, 2017. – Вип. 2(42). – С. 117-120.
20. Svyrydov, A., Kuchuk, N., Tsiara, O. (2018), "Improving efficiency of image recognition process: Approach and case study", Proceedings of 2018 IEEE 9th International Conference on Dependable Systems, Services and Technologies, DESSERT 2018, pp. 593-597, DOI: <http://dx.doi.org/10.1109/DESSERT.2018.8409201>
21. Кучук Г. А. Метод синтезу інформаційної структури зв'язного фрагменту корпоративної мультисервісної мережі / Г. А. Кучук // Збірник наукових праць Харківського університету Повітряних сил. – 2013. – № 2(35). – С. 97-102.
22. Kosenko V. Mathematical model of optimal distribution of applied problems of safety-critical systems over the nodes of the information and telecommunication network. *Сучасні інформаційні системи (Advanced Information Systems)*. 2017. Т. 1, № 2. С. 4-9. DOI: <https://doi.org/10.20998/2522-9052.2017.2.01>

Received (Надійшла) 25.12.2019

Accepted for publication (Прийнята до друку) 22.01.2020

Synthesis of computer system network model on a hyperconverged platform

N. Kuchuk

Abstract. The purpose of the article is to develop a method for synthesizing network models based on Petri time networks. The developed model will allow to build a model that allows to achieve the degree of adequacy for forecasting the performance of the software complex with the required reliability. **Research results.** A method of synthesis of a Petri time network based on data tracing is proposed. This method was used in modeling the process of functioning of a complex of programs. The main disadvantage of the proposed approach is the need to constantly collect measurement information in a computer system. It is shown that the degree of adequacy of the model depends on the accuracy of the measurement information. The adequacy of the description of the dynamics of the studied process is analyzed. To evaluate the effectiveness of the proposed model, it was used to predict the performance of the composite package used in airline sales. **Conclusions.** In practical applications, in most cases, it is sufficient to synthesize a network model of a computer system on a hyper-converged platform based on 30-35 measurements. Selecting a larger value to reduce the size of the confidence interval of the estimates obtained will not be justified if the prediction task is to minimize the prediction error.

Keywords: hyperconverged platform; model, Petri net.