

**В. М. Кичак, д. т. н., проф.; М. В. Васильська**

## **РОЗРОБКА УЗАГАЛЬНЕНОЇ ІМІТАЦІЙНОЇ МОДЕЛІ СИСТЕМИ МОБІЛЬНОГО ЗВ'ЯЗКУ НА БАЗІ АГРЕГАТНОГО ПІДХОДУ**

*Запропоновано і обґрунтовано побудову узагальненої, відкритої для модифікацій робочої математичної моделі багатоканальної системи масового обслуговування з довільними дисциплінами обслуговування. Функціональні підсистеми імітаційної моделі побудовані як агрегати. Цільове призначення моделі – моделювання процесів в системах мобільного зв'язку.*

**Ключові слова:** агрегат, агрегативна система, масове обслуговування, імітаційна модель, робоча модель, розподіл ймовірностей, невизначеності, векторизація, багатоканальна система.

**Постановка проблеми.** Системи мобільного зв'язку сьогодні є не тільки вагомим сектором індустрії і технологій, але і невід'ємною складовою усіх організаційно-виробничих систем. Тому безумовно актуальними є задачі прогнозування і планування розвитку систем мобільного зв'язку. Високі темпи інновацій, високі рівні невизначеностей, відсутність статистики обумовлюють необхідність використання математичних моделей функціонування таких систем. Одна з важливих задач в моделюванні систем мобільного зв'язку – отримання функцій впливу зовнішніх і внутрішніх факторів на ефективність систем мобільного зв'язку. Існуючі методи розрахунку черг та пропускнуєї спроможності систем мобільного зв'язку вимагають знання теоретичних частотних розподілів для інтервалів та обсягів надходження заявок (задач) і тривалості часу обслуговування.

**Постановка задачі.** Об'єднуємо етапи розробки математичної моделі та програми моделювання в розробку робочих моделей. Робоча модель – це продукт сучасних пакетів для моделювання взагалі та спеціалізованих пакетів для моделювання систем мобільного зв'язку. Робоча модель є фактично інтерфейсом користувача з програмними модулями пакета, написаними на мові C++. Візуальне подання робочої моделі досить точно відповідає стандартам наукових, зокрема, математичних публікацій.

### **Обґрунтування методу визначення функцій впливу**

В роботі вибирається базова модель функціонування системи мобільного зв'язку, як багатоканальної системи масового обслуговування (СМО) з довільними і змінними дисциплінами обслуговування. В рамках відомих аналітичних моделей СМО неможливо задовільно відобразити специфічні особливості та складність систем мобільного зв'язку. Вибрано шлях розробки імітаційної моделі, що відповідала би вимогам обчислювальної ефективності, модульності і придатності до модифікацій. В якості засобів реалізації моделі вибрані доступні математичні пакети Matlab, Mathcad, VisSim.

На сьогодні ефективним підходом до імітаційного моделювання складних багатоканальних систем є розбиття моделі СМО на агрегати і побудова агрегативних моделей. Агрегат є узагальненням елемента з теорії динамічних систем. Суть узагальнення в тому, що в складі агрегату можуть бути неперервні статичні і динамічні моделі, імпульсні, релейні, логіко-динамічні моделі, моделі кінцевих автоматів і моделі невизначеностей і випадкових збурень. Зокрема, це можуть бути класичні імовірнісні моделі і моделі на базі нечіткої логіки. Агрегат має входи і виходи, виступає як універсальний перетворювач інформації [1].

Модель складної системи, зібрана з агрегатів – агрегативна модель системи, отримується методами декомпозиції та агрегування. В середовищах математичних пакетів агрегат можливо визначити як функцію користувача, що бере вектори параметрів агрегату і вхідних змінних і повертає вектор вихідних змінних. Також можливо ефективно реалізувати функції оптимізації, пошуку нульових коренів і станів рівноваги.

Після визначення і реалізації агрегатів та агрегативних моделей можна оперувати з ними як з звичайними аналітичними функціями (наприклад  $\sin()$ ,  $\arcsin()$ ,  $\ln()$ ) – інтегрувати їх, шукати екстремуми і нульові точки, з певними обмеженнями знаходити похідні. Тобто, коли імітаційна агрегативна модель побудована, вона дозволяє просто знаходити функціональні залежності показників ефективності.

Крім декомпозиції моделі на елементи-агрегати, застосовуємо декомпозицію складної імітаційної моделі в часі: створюємо модель неодномоментно, а крок за кроком. На кожному кроці існуюча працездатна модель незначно ускладнюється, уточнюється і проходить широке тестування. На моделі також виконуються дослідження – згідно цільовому призначенню. В практичному аспекті покрокове створення моделі дозволяє уникати помилок і, головне, дозволяє отримувати нові знання про властивості об'єкта моделювання, коректувати напрямки розвитку імітаційної моделі.

При розробці чергової моделі, попередня модель залишається як засіб контролю більш складних моделей і як спрощена модель для спрощених задач. На базі розробленої моделі можуть створюватись імітаційні моделі конкретних систем мобільного зв'язку.

### Визначення базових понять

**Агрегатні моделі.** В літературі з моделювання не існує єдиних стандартизованих класифікацій моделей. Не існує також єдиного визначення агрегатних моделей та агрегативних систем, тому уточнюємо і конкретизуємо ці визначення стосовно об'єктів дослідження і моделювання – систем мобільного зв'язку. Розглянемо клас складних систем, що є конструкціями з агрегатів, що мають такі властивості: існує таке, звичайно не єдине розбиття системи на елементи, при якому кожним отриманим елементом є агрегат [2].

**Поняття агрегату в теорії систем.** Найбільш загальне визначення агрегату: задана  $T$  – множина моментів часу, множини довільної природи  $X, U, Y, Z$ . Елементи вказаних множин мають такі інтерпретації:  $t \in T$  – момент часу;  $x \in X$  – вхідний сигнал;  $u \in U$  – сигнал управління;  $y \in Y$  – вихідний сигнал;  $z \in Z$  – стан агрегату. Стани, входи, виходи, управління є функціями часу  $z(t)$ ,  $x(t)$ ,  $u(t)$  і  $y(t)$ .

Агрегатом називається об'єкт  $\langle T, X, U, Y, Z, H, G \rangle$ , де  $H, G$  – оператори, можуть бути випадковими, нечіткими. Виходами операторів переходів і виходів  $H$  і  $G$  є функції  $z(t)$  і  $y(t)$ . Відмінність агрегатів від кінцевих автоматів і звичайних динамічних систем в структурі цих операторів.

**Оператори переходів і виходів.** В напрямку «агрегативні імітаційні моделі» існують два підходи – абстрактний, де розглядаються класи операторів з «особливими станами» з штучними прикладами без змістовних інтерпретацій, і підхід, де головна мета – створення ефективних програм імітаційного моделювання для досліджень і вбудовування в автоматизовані системи управління. Послідовність етапів дослідження в останньому підході – створення працездатних моделей, проведення інтенсивних досліджень на моделі, потім теоретичне узагальнення для створення чергового покоління імітаційних моделей. Ні в реальних об'єктах, ні в моделях не повинно бути «особливих станів» тимчасової чи остаточної непрацездатності. Вибираємо саме такий підхід.

**Процес функціонування агрегату.** Агрегат є математичною схемою досить загального вигляду, окремими випадками якої є функції алгебри логіки, релейно-контактні схеми, кінцеві автомати, всі класи систем масового обслуговування, динамічні і логіко-динамічні системи, що описуються звичайними диференціальними рівняннями і деякі інші об'єкти. З погляду моделювання агрегат виступає як достатньо універсальний перетворювач, він сприймає вхідні сигнали, що управляють, і видає вихідні сигнали. В даній роботі набір математичних агрегатів розширюється за рахунок імітаційних моделей складних випадкових подій і моделей нечіткої логіки.

**Агрегативні системи.** Розглянемо клас складних систем, що є конструкціями з агрегатів, що володіють наступною властивістю: існує таке (у загальному випадку неоднозначне)

розбиття системи на елементи, при якому кожний отриманий елемент є агрегатом. Такого роду складні системи називатимемо агрегативними або А-системи. Бажано, але не обов'язково, в реальній системі поданій агрегативною моделлю мати спеціалізовані агрегати для обміну інформацією з зовнішнім оточенням. Відповідно, вся інформація, циркулююча в А-системі, ділиться на зовнішню і внутрішню.

Для кожного агрегату описуються умови передачі інформації – запізнення, передача за запитом, наявні шуми і збурення. Щодо структури агрегатів, то всі вони описуються теорією графів. Властивості А-системи визначаються не тільки властивостями агрегатів, але також і її структурою. Нас зв'язки між агрегатами цікавлять тільки в аспекті заміни певних структур еквівалентними агрегатами. Природно виділити послідовні, паралельні поєднання і цикли – зворотні зв'язки. Завжди можливо замінити такі структури еквівалентними агрегатами. Це твердження просто перевірити на рівні програмної реалізації – об'єднанням програмних модулів для агрегатів, що складають структури. Теоретичне доведення твердження можливе, але спряжене з великим обсягом процедур формалізації сценаріїв функціонування агрегатів – запізнь, зміни станів, збурень та ін.

Побудуємо базову агрегативну модель СМО, як інтерпретацію теоретичних положень.

### Вибір технології реалізації робочої моделі

Вибираємо технологію розробки базової моделі, максимально узгоджену з технологією розробки програмного забезпечення, – вибираємо для опису СМО підхід на базі "вектору стану" – вибираємо структуру вектору стану, що є компромісом між зручністю, точністю опису системи і розмірністю; – виконуємо ієрархічну декомпозицію моделі на малі функціональні модулі, кожний модуль ретельно тестуємо, перш ніж включати до головної програми; – модулі імітації поведінки формуємо як оператори перетворення поточного стану в наступний; – при переході від одновимірних систем до багатовимірних використовуємо векторизацію обчислень [3].

В порядку узагальнення це така послідовність моделей:

–  $x_{k+1} = A \cdot x_k + B \cdot u_k$  – лінійна система, об'єкти – вектори стану;

–  $Mu_{k+1} = Mu_k + Nast(Mu_k, dMu_k)$  – нелінійна система, об'єкти – матриці стану;

–  $Mu_{k+1} = Op(M_k, U_k, P)$  – нелінійна система, об'єкти – довільні матричні структури,  $U_k, P$  – управління і параметри.

Довільна матрична структура – матриця, елементи якої довільні матричні структури: скаляри, матриці з числовими і символічними змінними. В імітаційні моделі систем мобільного зв'язку необхідно включати моделі поведінки користувачів. Як приклад векторизації розглянемо модуль нечіткого вибору з навчанням, що бере скалярні параметри певного елемента і повертає скалярний параметр імовірність вибору. Спочатку робимо скалярний модуль.

$$dmu(vyb, bolv, nvz) = \begin{cases} kys \leftarrow vyb \cdot nvz \\ qq \leftarrow vyb \cdot bolv + rnd(kys) - 0.5 \cdot kys \\ qq \end{cases} \quad (1)$$

Застосовуємо модуль (1) до обробки масиву матричних об'єктів – масивів параметрів цих споживачів (статистика користування послугами мобільного зв'язку для кожного користувача), на виході теж отримуємо масив вихідних даних:

$$\xrightarrow{\hspace{10em}} dMU = dmu(Vyb, Bolv, Nvz) \quad (2)$$

### Розробка агрегатних модулів СМО

Ставимо дві робочі цілі розробки: створення відкритої бібліотеки модулів для різних Наукові праці ВНТУ, 2010, № 1

дисциплін обслуговування потоку вимог обслуговування, створення відкритої бібліотеки модулів імітації невизначеностей. Придатні для сучасних умов алгоритми обслуговування повинні бути простими - "природними", децентралізованими, відмовостійкими, адаптуємими до статистики вхідного потоку задач.

**Модуль "вибір розподілів імовірностей".** Модуль базується на функціях, що генерують масиви чисел з заданим розподілом імовірностей. Передбачено параметричне настроювання цих функцій на статистику реальних вхідних даних. На рис. 1 подано приклад реалізації бібліотеки невизначеностей.

$$kk := 3 \quad Fvx(kk) := \begin{cases} rbinom(1, 8, 0.9) & \text{if } kk = 1 & Fvx(1) \rightarrow rbinom(1, 8, .9) \\ rpois(x, \lambda) & \text{if } kk = 2 & Fvx(2) \rightarrow rpois(x, \lambda) \\ rnorm(1, 6, 1) & \text{if } kk = 3 & Fvx(3) \rightarrow rnorm(1, 6, 1) \end{cases}$$

Рис. 1. Бібліотека невизначеностей (частина)

**Модуль імітації невизначеностей.** На рис. 2 подано приклад модуля імітації потоку заявок на обслуговування – три випадкових параметра: кількість, обсяги (тривалості) і пріоритети обслуговування.

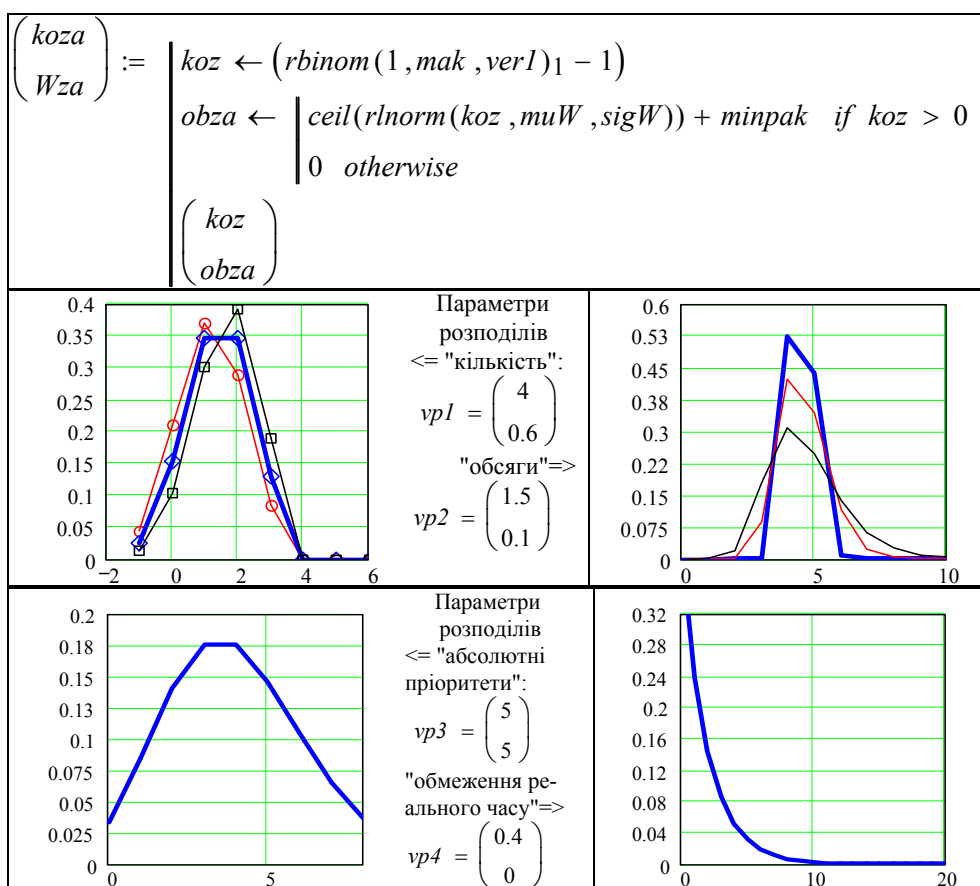


Рис. 2. Модуль імітації потоку заявок на обслуговування

В класичній літературі з СМО подаються детальні класифікації за типами розподілів ймовірностей вхідних потоків задач та тривалостей обробки. Причина цього в тому, що шукались аналітичні розв'язання індивідуально для кожного виду розподілу. Зроблений модуль генерації вхідного потоку задач може бути настроєний на довільні розподіли ймовірностей.

### Модулі формування стану черги та пріоритетів

Радикально розв'яжемо проблему вибору оптимального управління пріоритетами задач в черзі: сформуємо пріоритет як зважену суму від пріоритетів, визначених за різними алгоритмами, що звичайно використовують в обчислювальних системах [4].

$$Ps_i = \sum_{p=1}^{Np} \alpha_p \cdot prior_{i,p}, \quad (3)$$

де  $\alpha_p$  – ваговий коефіцієнт по  $p$ -тому показнику пріоритету  $prior_{i,p}$  – пріоритет  $i$ -ої задачі по  $p$ -тому показнику пріоритету. Структура даних "черга з пріоритетами" складається з вхідних даних і обчислюваних даних – пріоритетів. Подаємо чергу матрицею, де:

- перший стовпець – індекси "натуральної черги";
- другий стовпець – обсяги задач в натуральному порядку;
- третій стовпець – очікування в черзі (кроків, тактів);
- четвертий – обмеження реального часу (максимальне запізнення обробки);
- п'ятий – **абсолютні пріоритети**, що інтерпретуються як важливість задачі.

В стовпцях 6, 7, 8 розташовані **відносні пріоритети**, що обчислюються і можуть змінюватись на протязі очікування в черзі:

- 6 – пріоритет розміру задачі: чим менше розмір задачі, тим вище пріоритет;
- 7 – пріоритет очікування в черзі: чим довше очікування, тим вище пріоритет;
- 8 – пріоритет реального часу: чим менше часу до моменту обробки заявки, тим вище пріоритет задачі.

В останньому стовпці 9 – сумарний зважений пріоритет, що обчислюється за (3).

Зважена згортки (3) має ще одне призначення: обнуливши всі, крім одного, коефіцієнти, можемо реалізувати і досліджувати «в чистому виді» СМО з одним певним пріоритетом. Наявність "бібліотеки пріоритетів" утворює "базу знань" для побудови адаптивних систем.

**Динаміка пріоритетів.** Певні дані в матриці  $Ps\theta$  залежать від інших, або змінюються в часі згідно з певними механізмами. Класифікуємо ці зміни [5]:

- 1) зникнення рядків черги в результаті обробки задач згідно пріоритетам;
- 2) поява нових рядків при надходженні нових задач з відповідними пріоритетами;
- 3) переобчислення пріоритетів на кожному кроці процесу: збільшення очікування в черзі, збільшення пріоритетів для задач реального часу.

В підсумку нам потрібно отримати оператор переходу між поточним і наступним станами структури даних "черга з пріоритетами". Спочатку зробимо модулі, що описують перетворення окремих фрагментів структури даних – стовпців, рядків.

**Модуль обчислення пріоритетів за розмірами задач** бере другий стовпець  $Ps\theta^{(2)}$  (обсяги задач) структури даних, а повертає нормований вектор пріоритетів.

**Модуль обчислення середнього зваженого пріоритету** бере стовпці 5 – 8 матриці  $Ps\theta$ , а повертає середній зважений вектор пріоритетів, що розміщується в останньому стовпці матриці  $Ps\theta$ .

Результатом роботи розглянутих вище модулів є формування стовпців 6 – 9. В результаті формується структура зручна для програмної реалізації широкого класу алгоритмів управління процесом обробки потоку задач.

**Модуль обробки черги.** Агрегатна модель обробки черги крім функції визначення пріоритетів та вибору чергової задачі для обробки, повинна з кожним кроком обробки впорядковувати матрицю стану черги. Це можна зробити по різному: обнулити оброблені рядки, або обрізати, або ігнорувати, сортувати за різними змінними, а потім сортувати назад. Звичайно буде вибрано найгірший варіант, якщо не йти кроками "модуль менше семи рядків" і не перевіряти кожний крок. Випробувано декілька альтернатив і обґрунтована експериментами останню альтернативу. Лінгвістична модель процесу поповнення і обробки

черги:

- на кожному кроці надходять нові задачі (вихід модуля імітації потоку задач);
- на кожному кроці обробляється черга задач в обсязі пропускної спроможності.

Можливі ситуації обробки черги:

- 1) поточна черга = 0; ==> наступна черга = 0;
- 2) "поточна черга" ≤ "пропускна спроможність" ==> наступна черга = 0;
- 3) пропускна спроможність < обсяг першої задачі ==> залишок першої задачі в наступному кроці ще буде в обробці;

Кінцева ціль розробки програми моделювання – зробити таку структуру і такий оператор (функцію користувача), щоб динаміка системи описувалась таким різницеvim рівнянням перетворення стану системи:

$$ss^{(k+1)} = Y_{pur} P \left( ss^{(k)} \right), \quad (4)$$

$$\text{або } ss^{(k+1)} = Y_{pur} P I \left( ss^{(k)}, u \left( ss^{(k)} \right) \right). \quad (5)$$

Модуль обробки черги бере пропускну спроможність СМО та вектор сортованої за вибраними пріоритетами черги задач, а повертає новий вектор черги (це відображено в означеннях  $ss3$ ,  $S3$  – означення відповідного компонента структури стану ОС в головній робочій моделі). Словесний опис процесу обробки черги:

- робимо "заглушку" для випадку нульової черги:  $nulka \leftarrow (0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0)$ ;
- визначаємо кількість заявок  $Okol$  та обсяг  $Odlin$  черги;
- сортуємо матрицю стану черги по вибраному пріоритету (змінна  $u$ );
- формуємо умову "черга не нульова".

На рис. 3 подано текст модуля, що реалізує цей словесний опис.

$smHP(props, ss3, u) :=$	$nulka \leftarrow (0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0)$ $Okol \leftarrow rows(ss3)$ $srti \leftarrow csort(ss3, u)$ $Odlin \leftarrow \sum_{i=1}^{Okol} srti_{i,2}$ $yslo \leftarrow Odlin > props$ $S3 \leftarrow \begin{cases} sjelo(props, srti, u) & \text{if } yslo \\ nulka & \text{otherwise} \end{cases}$ $S3$
--------------------------	---

Рис. 3. Агрегатна модель «модуль обробки черги»

Аналогічно були розроблені агрегатні моделі для наступних функціональних модулів.

**Модуль поповнення черги**, що приймає в чергу нові задачі, обчислює значення відносних пріоритетів задач, обчислює динамічні показники задач черги: очікування в черзі, критичний час для задач реального часу.

**Модуль динаміки пріоритетів**, що відстежує і обчислює змінні відносні пріоритети.

**Модуль "оператор переходу для СМО з пріоритетним обслуговуванням"**. Цей модуль використовує попередні модулі-агрегати, схема зв'язків яких подана на рис. 4. Словесний опис процесу переходу від попереднього стану в наступний:

- визначається результат обробки попередньої черги  $S30 \leftarrow smH(props, ss3)$ ;
- на кожному кроці обробляється черга задач в обсязі пропускної спроможності;
- визначаються довжина  $S4 \leftarrow length(S30)$  і обсяг  $S5 \leftarrow mean(S30) \cdot S4$  черги після обробки;
- за допомогою модуля "поповнення черги":  $BY \leftarrow koWza(vp1, vp2)$  визначається кількість  $S1 \leftarrow BY_1$  і обсяг  $S2 \leftarrow BY_2$  нових задач, що прийшли на протязі кроку процесу;
- формується черга на наступний крок  $S31 \leftarrow stack(S30, S2)$  (нові задачі ставляться в "хвіст" черги);
- формується вихід – наступний вектор стану обчислювальної системи.

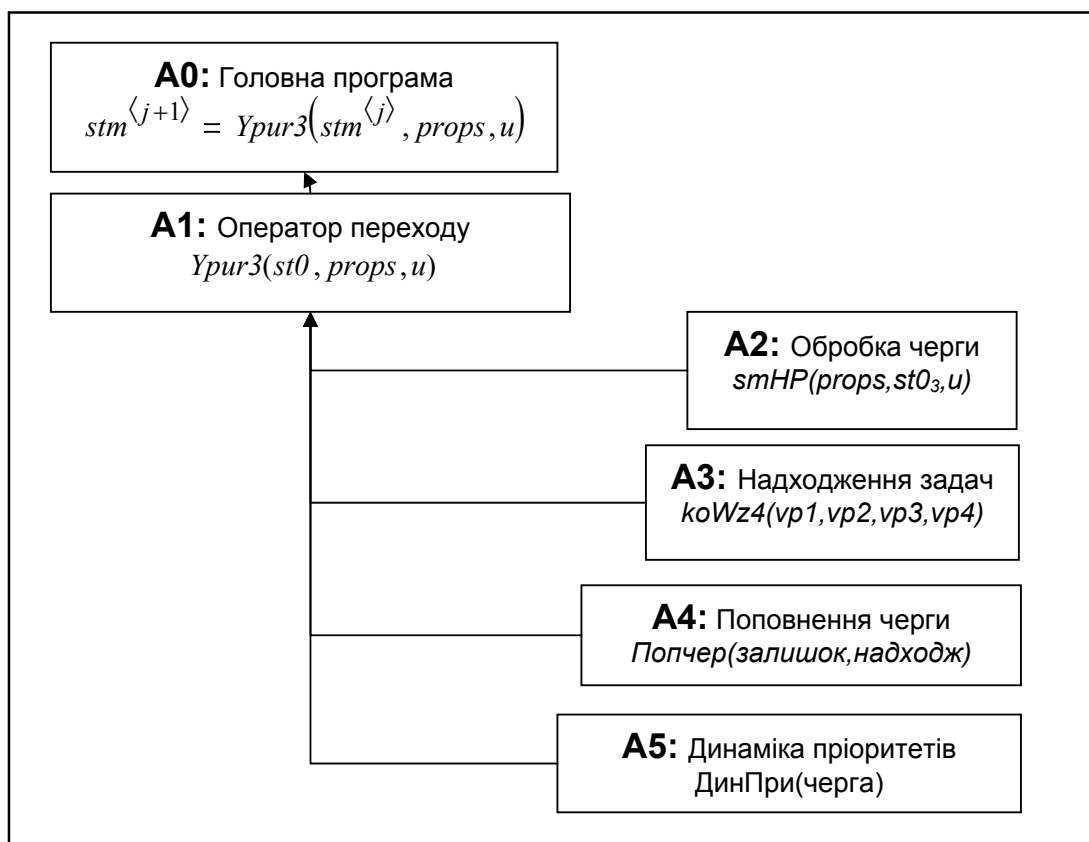


Рис. 4. Структура агрегативної моделі «система масового обслуговування з довільними пріоритетами і довільними розподілами параметрів обслуговування»

Для даної задачі існувала альтернатива – зроблена за традиційної технологією програма моделювання. Вона не була доведена із-за складності логіки роботи СМО в цілому.

Агрегативна імітаційна модель була доведена до працездатного стану і на ній проведено великий обсяг досліджень. На рис. 5 та 6 подано два приклади моделювання процесів обслуговування з різними пріоритетами.

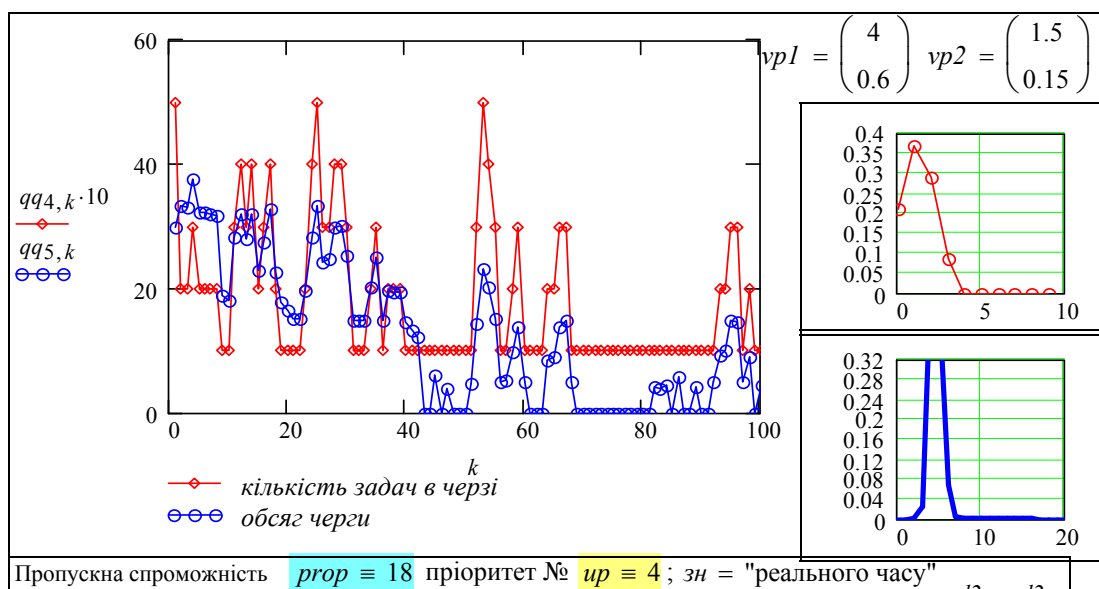


Рис. 5. Приклад результатів моделювання. Процес для пріоритету обслуговування «реальний час»

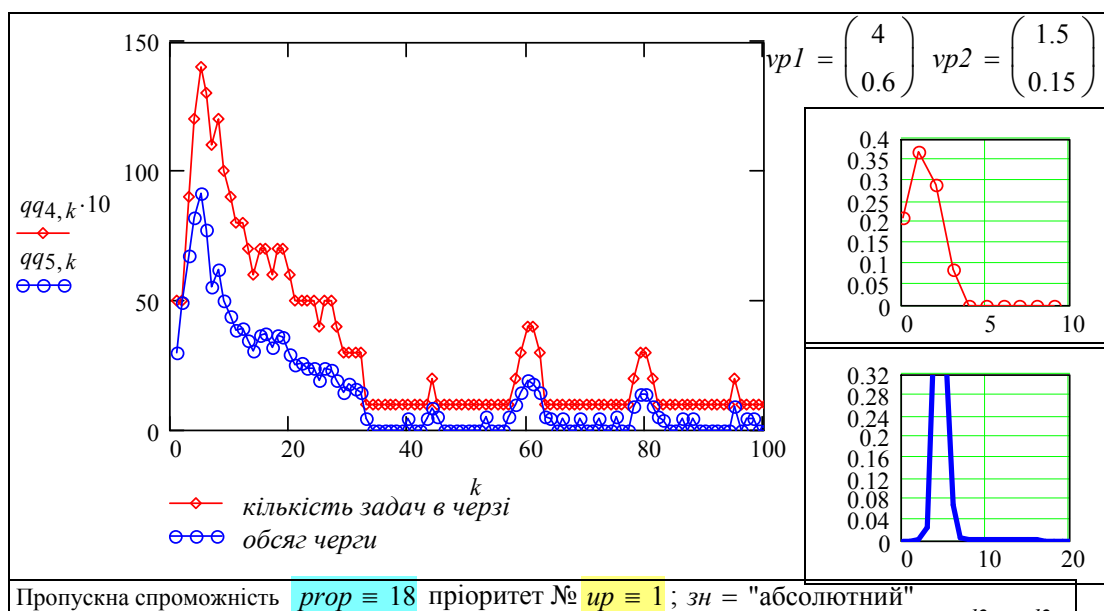


Рис. 6. Приклад результатів моделювання. Процес для пріоритету обслуговування «абсолютний»

**Практичне значення** розробки моделі й проведених досліджень – визначення напрямків збору даних у реальних системах мобільного зв'язку для налаштування і модифікації даної імітаційної моделі, створення моделі-предиктора для реальних систем.

**Теоретичне значення** – визначення напрямків теоретичного обґрунтування виявлених властивостей системи масового обслуговування, узагальнення теоретичних моделей систем масового обслуговування. Перспективним напрямком є отримання і аналіз статистики віртуальної реальності та отримання функцій впливу.

#### СПИСОК ЛІТЕРАТУРИ

1. Нэгл Томас Т. Стратегия и тактика ценообразования. Руководство для принятия решений, приносящих прибыль / Томас Т. Нэгл. – М.: Питер, 2001. – 375 с.
2. Kelly K. New Rules for the New Economy. 10 radical strategies for a connected world. – Penguin books, 1999. – 180 p.
3. Форрестер Дж. Основы кибернетики предприятия / Дж. Форрестер. – М.: Прогресс, 1971. – 340 с.



4. Боровська Т. М. Основи теорії управління та дослідження операцій. Навчальний посібник / Т. М. Боровська, І. С. Колесник, В. А. Северілов. – Вінниця: УНІВЕРСУМ-Вінниця, 2008. – 242 с. – ISBN 978-966-641-275-4.

5. Моделювання та оптимізація у менеджменті: Навчальний посібник / Т. М. Боровська, В. А. Северілов, С. П. Бадьора, І. С. Колесник. – Вінниця: УНІВЕРСУМ-Вінниця, 2009. – 145 с. – ISBN 978-966-641-287-7.

**Кичак Василь Мартинович** – д. т. н., професор кафедри телекомунікаційних систем і телебачення; інститут радіотехніки, зв'язку та приладобудування.

**Васильська Майя Валеріївна** – аспірант кафедри телекомунікаційних систем і телебачення; інститут радіотехніки, зв'язку та приладобудування  
Вінницький національний технічний університет.