


НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
"КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ"

На правах рукопису

ТОМАСЕВСЬКИЙ ВАЛЕНТИН МИКОЛАЙОВИЧ

УДК 681.51

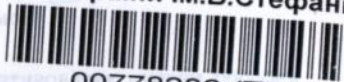


**МЕТОДИ ПРОЕКТУВАННЯ ІНТЕРАКТИВНИХ СИСТЕМ
ІМІТАЦІЙНОГО МОДЕЛЮВАННЯ**

Спеціальність 05.13.04 - автоматизовані системи
управління та системи обробки інформації

А В Т О Р Е Ф Е Р А Т
дисертації на здобуття вченого ступеня
доктора технічних наук

Київ - 1996



00778232 (Т)

Роботу виконано на кафедрі інформації та управління Національного технічного університету України "Київський політехнічний інститут"

- Науковий консультант: доктор технічних наук, професор Павлов О.А.
- Офіційні опоненти: доктор технічних наук, професор Вінницький В.П.
доктор технічних наук, професор Михайленко В.М.
доктор технічних наук, професор Святний В.А.
- Провідна організація: Інститут програмних систем НАН України, м. Київ

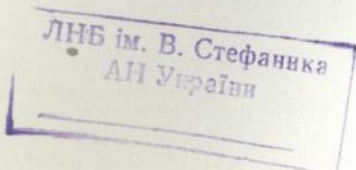
Захист дисертації відбудеться "18" 03 1996 р. о 15 годині на засіданні спеціалізованої Ради Д 01.02.08 при Національному технічному університеті України "КПІ" за адресою: 252056, м. Київ, проспект Перемоги, 37, ауд. 56-14.

З дисертацією можна ознайомитися в бібліотеці НТУУ "КПІ".

Автореферат розіслано "___" _____ 1996 р.

Учений секретар спеціалізованої Ради, доктор технічних наук

Романенко В.Д.



АНОТАЦІЯ

Метою роботи є розробка методології та на її основі методів проектування інтерактивних систем імітаційного моделювання (ICIM), які призначені для аналізу функціонування дискретних систем і процесів, що протікають з різними швидкостями.

Для досягнення поставленої мети в роботі розв'язані такі завдання:

1. Розроблені на основі теоретико-множиного опису процесів системні принципи побудови ICIM і метод синтезу програмних реалізацій імітаційних моделей. Обмеженнями використання об'єктів і систем для моделювання конкретної предметної області є уявлення їх класом стохастичних мереж.

2. Розроблена процесна схема подання дискретних динамічних систем, котра є найбільш загальною для побудови імітаційних моделей, їх програмних реалізацій і інтерактивних систем проектування.

3. Розроблено метод потокового агрегування для класу швидких регенеруючих процесів, який дозволяє зменшити розмір імітаційної моделі за рахунок використання параметричної аналітико-статистичної моделі.

4. Розроблено метод прискорення моделювання повільних, зі стаціонарними ймовірними характеристиками процесів з використанням авторегресійної моделі часового ряду та відновленими значеннями для прогнозованих рівнів вихідних змінних імітаційної моделі, який дозволяє встановити залежність між об'ємом вибірки та кореляцією даних.

5. Розроблено адаптивний алгоритм прискореного моделювання для класу регенеруючих процесів.

6. Побудована ICIM, яка працює разом з системою моделювання GPSS/PC і системою статистичної обробки STATGRAPHICS.

Автор захищає:

1. Системну методологію побудови ICIM, її архітектуру та програмну реалізацію.

2. Метод формалізації та синтезу програмних реалізацій імітаційних моделей класу стохастичних мереж з використанням схеми процесів і лінгвістичного процесора.

3. Метод потокового агрегування для регенеруючих процесів.
4. Метод прискореного моделювання процесів зі стаціонарними ймовірнісними характеристиками.
5. Адаптивний алгоритм прискореного моделювання для регенеруючих процесів.
6. Методику прийняття рішень з використанням імітаційних моделей.
7. Результати використання імітаційних моделей систем збору, передачі та обробки даних.

ЗАГАЛЬНА ХАРАКТЕРИСТИКА РОБОТИ

Актуальність проблеми. Імітаційне моделювання застосовується в усіх сферах діяльності людини: від моделей технічних, технологічних та організаційних систем до проблем розвитку людства та всесвіту.

Добре відомі роботи з імітаційного моделювання Р. Шенона, Дж. Шрайбера, Дж. Клейнена, А. Прицкера, перекладені російською мовою. Широке використання цього методу за кордоном не привело поки у нас в країні до його великого поширення і використання для управління, незважаючи на наявність фундаментальних робіт у цій області російською мовою, і робіт що проводилися Інститутом кібернетики АН України. Мають, це пояснюється перебудовою принципів мислення при розробці імітаційних моделей порівняно з традиційними програмними засобами.

Для вирішення завдань методом імітаційного моделювання створено значна кількість різних систем, більшість з яких є програмними, але розробка їх продовжується. Це можна пояснити удосконалюванням обчислювальної техніки і програмних засобів і включенням до нових систем різних оригінальних ідей.

Як об'єкт моделювання в роботі обрані стохастичні мережі систем масового обслуговування (СМО), які знайшли поширення при моделюванні систем обробки і передачі даних, виробничих, транспортних систем, служб ремонту та інших.

Імітаційне моделювання стохастичних мереж використовують у тих випадках, коли моделі не піддаються аналітичним розрахункам, або для підтвердження одержаних аналітичних розрахунків. При імітаційному моделюванні мереж виникають проблеми, пов'язані з моделюванням повільних процесів з рідкими подіями. Для одержування

надійних статистичних оцінок необхідно орієнтуватися на найповільніший процес у моделі, в той же час швидкі процеси будуть утворювати велику кількість подій у моделі, для обробки яких треба багато машинного часу. Для вирішення цієї проблеми використовують комбіноване моделювання з аналітичною та імітаційною або статистичною моделями.

Комбіновані методи припускають відомими функції розподілу для випадкових процесів, використовують різні методи зниження дисперсії шуканих оцінок, поєднують аналітичний розрахунок та імітація деяких подій за принципом усереднення на основі СМО або регресійного аналізу. Більшість цих методів пов'язано з постановками завдання моделювання, що перешкоджає використування їх для практичного імітаційного моделювання.

Для більш ефективного використання методу імітаційного моделювання необхідно розробити методологію синтезу імітаційних програм, яка повинна базуватися на формальному уявленні процесу проектування імітаційної моделі. Як формальні схеми використовують агрегати, СМО, ймовірнісні автомати або типові обчислювальні схеми імітації.

У роботі пропонується застосовувати схему процесів і лінгвістичний процесор для автоматизованої генерації імітаційних програм. Формальне уявлення в вигляді схеми процесів дозволяє не тільки зробити опис алгоритму моделювання, але і процесу роботи з користувачем в інтерактивній системі.

Аналіз різних засобів автоматизації та побудови програмних генераторів для імітаційного моделювання показав, що такі системи не зручні в користуванні, потребують значних ресурсів ЕОМ, засобів роботи з базами даних. Для реалізації діалога з користувачем використовуються спеціалізовані або близькі до природних мови. Перші потребують освоєння їх користувачем, а другі - складних схем трансляції. Крім того, ні одна з відомих автору систем не має засобів перетворення імітаційних моделей, які дозволяють зменшити її розмір і прискорити процес моделювання.

Розробка складних універсальних систем моделювання пов'язана зі значними витратами, крім того за універсальність завжди треба розраховуватися швидкістю моделювання та об'ємом потрібної пам'яті для системи. Тому, при розробки таких систем перевагу треба віддавати простим, надійним і найбільш ефективним з точки зору потреб до ресурсів ЕОМ засобам.

Дисертаційна робота спрямована на вирішення актуального завдання, пов'язаного з розробкою компактних, надійних і високоефективних з точки зору вимог до ресурсів інтерактивних систем імітаційного моделювання дискретних стохастичних мереж, які мають можливість використовувати користувачі-непрограмісти. Відзнакою такої системи від інших є те, що до неї залучені методи потокового агрегування та прискорення імітаційного моделювання.

Методи досліджень. Загальна методика проектування інтерактивних систем імітаційного моделювання базується на системному підході. В роботі також використан операційний аналіз, імітаційне моделювання, параметрична ідентифікація, аналіз часових рядів, теорія масового обслуговування і статистичний аналіз.

Наукова новина дослідження полягає у розробці системних принципів побудови ІСІМ і метода синтезу програмних реалізацій імітаційних моделей для дискретних стохастичних мереж на основі схеми процесів і лінгвістичного процесора.

Новими результатами є:

- метод потокового агрегування для класу швидких регенеруючих процесів, який дозволяє зменшити розмір імітаційної моделі шляхом укрупнення її станів за рахунок побудови параметричної аналітико-статистичної моделі і заміни послідовності блоків імітаційної моделі на один блок затримки;

- метод прискореного моделювання повільних, зі стаціонарними ймовірнісними характеристиками процесів з використанням авторегресійної моделі часового ряду з відновленими значеннями для прогнозованих рівнів вихідних змінних імітаційних моделей;

- адаптивний алгоритм прискореного моделювання для класу регенеруючих процесів;

- інтерактивна система імітаційного моделювання, яка включає до себе засоби потокового агрегування, прискореного моделювання та працює разом з системою моделювання GPSS/PC і пакетом статистичної обробки STATGRAPHICS.

Практична цінність результатів, які одержано в роботі, складається в тому, що отримані теоретичні результати було втілено в ІСІМ, яка дозволяє автоматизувати процес проектування програмних реалізацій імітаційних моделей. Розроблена у складі ІСІМ підсистема проведення експериментів має самостійне значення та дозволяє вирішувати задачі прискорення імітаційного моделювання та статистичного аналізу результатів.

Розроблені імітаційні моделі можуть використовуватися в системах обробки даних, обчислювальних системах і мережах.

Техніко-економічну доцільність використання імітаційних моделей підтверджено актами та документами про впровадження та використання ІСІМ в учбовому процесі.

Реалізація результатів роботи. Розроблена ІСІМ передана для розповсюдження до ООБ "Кобріз", впроваджена в фірмі "М - ИНФО" для розрахунків локальних мереж ШЕОМ, у ООБ "Лелія" для вибору найкращих рішень управління процесом пошиття одягу та використовується в учбовому процесі кафедри АСОІУ Національного Технічного Університету України "КПІ".

Імітаційні моделі знайшли використання при проектуванні обчислювального центра в м. Тула, інформаційнодовідкової системи Міністерства зв'язку УРСР в м. Києві, при проектуванні галузевої мережі обчислювальних центрів для ОАСУ Мінелектротехпрому СРСР в м. Москві.

Імітаційна модель галузевої системи збору, передачі, підготовки та обробки інформації використовувалась для аналізу роботи цієї системи у рамках АСУ-ГІОЦ Мінелектротехпрому в м. Москві.

Практичне використання результатів роботи підтверджується відповідними документами.

Апробація роботи. Основні положення роботи доповідались і обговорювались на конференціях і семінарах: Всесоюзному семінарі "Алгоритмізація в АСУ" (Київ, 1974 р.); Всесоюзній конференції "Математичне забезпечення моделювання складних систем" (Київ, 1977 р.); конференції Республіканського Дома економічної і науково-технічної пропаганди та Інституту кібернетики АН УРСР "Математичне забезпечення моделювання складних систем" (Київ, 1980 р.); республіканському семінарі "Системи і засоби інтегрованої обробки інформації" (Кишинев, 1981 р.); Галузевій науково-технічній нараді "Досвід розробки і впровадження систем обробки даних в електротехнічній промисловості" (Талін, 1982, 1986 р.); семінарі "Досвід побудови та експлуатації мереж ЕОМ" (Севастополь, 1988 р.); I, III Міжреспубліканській конференції "Розробка автоматизованих систем управління" (Київ, 1987 р., 1989 р.); Всесоюзній школі "Імітаційні експерименти в моделях складних систем" (Калінінград-областний, 1989 р.); семінарі "Автоматизація наукових досліджень та іспитів" ради по проблемі "Кібернетика" НАН України (Київ, 1994 р.); Міжнародних конференціях "Автоматика - 94" (Ки-

Ів, 1994 р.), "Укрсофт - 94" (Львів, 1994); Белоруській Міжнародній Зимовій школі-семінарі (BWWQ-96) "Дослідження систем і мереж масового обслуговування" (Гродно, 1996).

Публікації. Основні положення і результати досліджень по темі дисертації опубліковані в 30 наукових працях.

Структура та обсяг роботи. Дисертаційна робота складається з вступу, шести глав, заключення, списку літератури і додатків. Загальний обсяг роботи 266 сторінок машинописного тексту, в тому числі рис. 45, таб. 12, список літератури з 136 найменувань.

У вступі обґрунтовується актуальність теми, формулюється мета роботи, приводиться загальна характеристика.

У першій главі на основі системного аналізу вводиться формальне визначення поняття процесу, з допомогою якого розкривається внутрішній механізм організації алгоритмів моделювання, формулюється принципи розробки імітаційних моделей.

У другій главі з єдиних системних позицій розглядаються питання побудови імітаційних моделей та їх використання, розкриваються основні особливості імітаційного моделювання у вигляді сукупності етапів і робіт. Показано, що всі ці етапи взаємопов'язані. Проведений аналіз мов моделювання показав, що мови, орієнтовані на процеси, мають більш підходящі засоби для проектування складних імітаційних моделей з використанням принципів низхідного і об'єктного проектування програм.

У третій главі розглядаються питання проектування програмних реалізацій імітаційних моделей. Проводиться систематизація методів проектування, показано, що використання інтерактивних засобів генерації програм дозволяє скоротити час утворення імітаційних моделей і звільнити користувача від необхідності програмування моделей. Проведений системний аналіз з використанням теоретико-множиного опису проектуємої системи показав, що стохастичні мережі добре відображують структури концептуальної та імітаційної моделей.

У четвертій главі розглядаються питання проведення експериментів і прийняття рішень з допомогою імітаційних моделей, розкриваються проблеми статистичної обробки та оцінки точності результатів моделювання. З метою скорочування часу моделювання проводиться потокове агрегування, яке дозволяє зменшити розмір імітаційної моделі за рахунок укрупнення її станів.

Для прискорення моделювання повільних процесів використову-

ється авторегресійна модель часового ряду з відновленими значеннями для прогнозованих рівнів вихідних змінних імітаційної моделі. Приводиться адаптивний алгоритм для прискорення імітаційного моделювання регенеруючих процесів.

Процедури прийняття рішень ґрунтуються на формулюванні гіпотез і виборі найкращої з них. Методика автоматизованого прийняття рішень показана на прикладі імітаційної моделі виробничої ділянки.

У п'ятій главі розглядається методика побудови інтерактивних систем моделювання на основі мови GPSS. Розробляється програмний генератор з використанням лінгвістичного процесора, який дозволяє будувати синтактично і семантично вірні GPSS-програми. Приводиться система багатовіконного меню для інтерактивної системи моделювання і приклад роботи з нею.

У шостій главі приведені приклади реалізації розроблених методів при побудові і впровадженні різних імітаційних моделей систем обробки даних та інтерактивних систем проектування імітаційних моделей. Розглядаються імітаційні моделі багатотермінальних обчислювальних систем, систем збору, передачі і обробки інформації для ГАСУ.

У додатках приведені акти впровадження і додатковий матеріал для окремих глав.

ЗМІСТ РОБОТИ

Система S задається як пара $S = \langle X_S, R_S \rangle$, де X_S, R_S - множина елементів і відношень між ними. Розглянемо систему $Y R X(t)$.

Другий елемент відношення $X(t)$ є функцією часу, тобто, в свою чергу деякою множиною. Згідно з теоремою про розклад подамо R у вигляді добутку відношень

$$Y R_1 [X^j(t), Z^j], Z^j R_2 X^{jr}(t),$$

де Z стан системи в момент $\tau - t_j$; X, Y відповідно входи та виходи S .

Тоді рівняння виду

$$z(t > \tau) = z(z(\tau); x(\tau, t)) \quad (1)$$

називається рівнянням станів системи, а функція z - перехідною функцією станів системи.

Якщо ввести множину T як певну впорядковану множину позитивних дійсних чисел t , які визначають хід часу, то пару $\langle t, z \rangle$, де

$t \in T$, $z \in Z$, називають станом або фазою системи S , а множину $T \times Z$ - простором станів або фазовим простором системи. Перехідна функція z або її графік у просторі станів визначає поведінку системи або її траєкторію на деякому інтервалі часу $t \in [t_0, t_1]$.

Отже, щоб відтворити поведінку системи або її траєкторію необхідно задати рівняння станів системи (1). В моделюванні це рівняння також називають функцією дії.

Тоді процес, який описує поведінку системи, визначається послідовністю станів, зв'язок між якими задається функцією дії та початковим станом системи. Отже, послідовність упорядкованих у порядку зростання часу пар $\langle z(t), x(\tau, t) \rangle$ визначає процес і описує поведінку системи.

На рис. 1 зображено діаграму станів СМО з одним приладом обслуговування. На вхід системи надходять заявки в моменти часу t_j^B , ($j=1, \dots, 8$). Момент часу початку обслуговування приладом позначено як t_j^N , а момент закінчення обслуговування - t_j^K . Момент часу виходу заявки із системи $t_j^{Вих}$ збігається з моментом закінчення обслуговування заявки приладом t_j^N . Якщо в момент надходження заявки в систему прилад обслуговування зайнятий, то вона змушена чекати, як, наприклад, заявки з номерами 2,4,5,6,7,8. З заявок, які чекають, утворюється черга. Стан черги визначається числом заявок та позицією кожної із них у черзі. Під час звільнення приладу перша заявка з черги надходить на обслуговування у прилад. Отже, простір станів СМО визначається станами приладу та черги. Зміна станів пов'язана з подіями надходження нової заявки та заняття або звільнення приладу обслуговування. Однак стан у черзі залежить від стану приладу обслуговування. Подіями для заявки j є поява заявки - c_j^B , початок обслуговування - c_j^N , кінець обслуговування - c_j^K .

Оскільки будь-яка послідовність станів, упорядкована в часі, є процес, то обслуговування заявки в СМО визначається як послідовність станів СМО від моменту надходження заявки - t_j^B до моменту t_j^K , коли вона покидає систему.

Перевірка умови u_3 зайнятості приладу в момент приходу заявки j визначає можливість її негайного обслуговування. Якщо прилад зайнятий, то заявка змушена чекати деякий час $t_j^4 = t_j^N - t_j^B$ (функція дії ЧЕКАННЯ - $d(t_j^4)$), яка зв'язана з умовою зайнятості приладу u_3). Час обслуговування визначається як $t_j^{06} = t_j^K - t_j^N$ (функція дії ОБСЛУГОВУВАННЯ - $d(t_j^{06})$), яка зв'язана з умовою

звільнення приладу $u_{зв}$). Загальний час перебування в СМО

$$t_j^{пер} = t_j^ч + t_j^{об}.$$

Тоді тривалість часу чекання в черзі є функцією дії для черги, а тривалість обслуговування приладом залежить від функції дії приладу. Упорядковані пари $\langle t_j^ч, t_j^{об} \rangle, j=1, \dots, 8$ визначають процес обслуговування заявки в СМО.

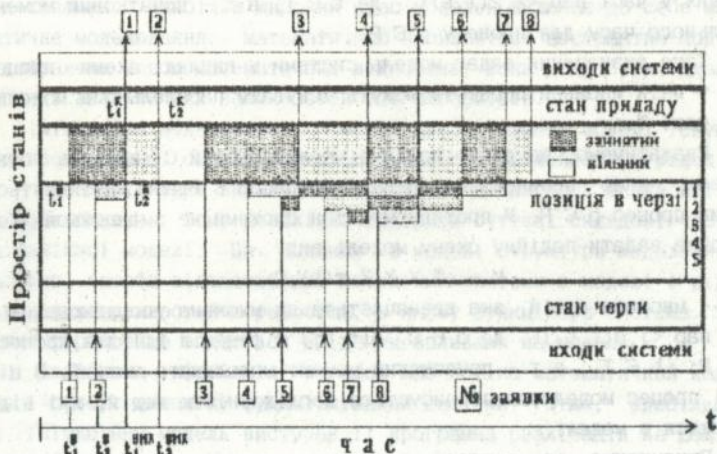


Рис. 1

Аналогічно, наприклад, можливо визначити процес появи заявок у СМО як упорядковані пари $\langle j, t_j^{появ} \rangle$, де j номер заявки яка надійшла, а $t_j^{появ} = t_j^в + \Delta t_j^{в+1}$ (функція дії ПРИХІД - $d(t_j^{появ})$, яка пов'язана з умовою $u_{поя}$ появи заявки).

Умовою початку дії є умова перевірки стану системи (прилад зайнятий або вільний, надійшла заявка чи ні).

Отже, процес починається тільки тоді, коли змінився стан системи та виконується функції дії. Зміна стану визначається настанням певної події, а початок виконання функції дії визначається здійсненням певних умов.

Тоді впорядкована множина подій $S = \{c_j^в, c_j^п, c_j^к\}, j=1, 2, \dots$ у модельному часі t описує поведінку СМО.

Якщо щоразу при зміні модельного часу перевіряти множину умов $U = \{u_{поя}, u_з, u_{зв}\}$ для кожної заявки j , щоб почати деяку дію із множини дій $D = \{d_j^{появ}, d_j^ч, d_j^{об}\}, j=1, 2, \dots$, то пари $\langle U, D \rangle$ теж описують поведінку СМО.

Формальний аналіз систем дає змогу визначити модель динамічної системи як структуру вигляду

$$M = \langle T, X, Y, Z, z(t), P \rangle, \quad (2)$$

де T - модельний час; X, Y - множина відповідно вхідних і вихідних змінних; Z - простір станів моделі; $z(t)$ - функція станів, $t \in T$; P - множина процесів, яка визначається як множина упорядкованих у часі пар $\langle x, z(\tau, t) \rangle$, де $t \in T$, а τ - початковий момент модельного часу для процесу $p \in P$.

Таке визначення задає модель системи у вигляді схеми процесів, у якій множини процесів можуть існувати паралельно в модельному часі T .

Вважатимемо, що деяка подія з множини подій C змінить стан системи, якщо починається певний процес $p \in P$ або закінчується деякий процес $p \in P$. У протилежному стан системи не змінюється. Тоді можна задати подійну схему моделі виду

$$M = \langle T, X, Y, Z, z(t), C \rangle, \quad (3)$$

де C - множина подій, яка визначається як множина упорядкованих у часі пар $\langle j, d(\tau, t_j) \rangle$, де $c \in C$, $d(\tau, t_j)$ - функція дії для процесу $p_j \in P$; $t \in T$, а τ - початковий момент модельного часу T . В цій схемі процес моделювання описується послідовністю подій, які відбуваються в моделі.

Припустимо, що завдяки виконанню деякої умови u з множини умов U почне виконувати певну дію $d(\tau, t_j)$ з множини дій D для деякого процесу $p_j \in P$. Тоді можливо задати модель системи у вигляді схеми дій

$$M = \langle T, X, Y, Z, z(t), D \rangle. \quad (4)$$

В цій схемі процес моделювання описується, як перевірка всіх умов при кожній зміні модельного часу $t \in T$, щоб знайти умову, яка почне певну дію з множини дій D . Зміна часу t може відбуватися з постійним кроком або від події до події.

Якщо припустити, що виконання деякої множини процесів P може привести до зміні станів $z \in Z$ і появу нових процесів, що спричинить появу деякої множини ситуацій L , тобто $Z(t): Pz \rightarrow L$, то отримуємо ситуаційну або причинно-наслідкову схему

$$M = \langle T, X, Y, Z, z(t), L \rangle. \quad (5)$$

В цій моделі необхідно описати множину ситуацій та множину правил (алгоритмів), що визначають який процес повинен виконуватись. Поведінка моделі в таких системах визначається ланцюгом виду

$$\{ \text{ситуація} \} \rightarrow \{ \text{правило} \} \rightarrow \{ \text{процес} \}.$$

Якщо модель має можливостями конструювання нових правил з існуючих, то така модель перетворюється на модель зі штучним інтелектом.

Існує деяка неоднозначність у термінології моделювання, зокрема, це стосується математичного, імітаційного та статистичного моделювання. Найбільш широкий клас моделей включає до себе математичне моделювання. Математичною називається абстрактна модель, відображена мовою математичних відношень. Вона може бути задана у вигляді аналітичних залежностей або алгоритмів.

Імітаційне моделювання - метод конструювання моделі існуючої або проєктованої системи та проведення експериментів на моделі.

Однак під таке визначення практично підлягають майже всі види моделювання. Тому необхідно виділити суттєві складові частини імітаційної моделі. Це подання в моделі структури модельованої системи, засоби відтворювання поведінки системи в моделі і відображення властивостей середовища, в якому функціонує система. Зовнішнє середовище задається вхідними впливами на модель.

Зазначена інформація загалом має логіко-математичний характер і описує динаміку функціонування системи. Отже, здебільшого за імітаційну модель виступає її програмна реалізація на ЕОМ, а імітаційне моделювання зводиться до проведення експериментів з моделлю шляхом прогонів програми на деякій множині даних, що є середовищем системи.

З більш математичної точки зору імітаційну модель можна розглядати як складена з рівнянь, які розв'язуються просліджуванням еволюції їх розв'язків на певному відрізку часу, тобто хронологічної послідовності подій.

Імітаційна модель може бути детермінованою або стохастичною. За наявності в моделі випадкових факторів виникає необхідність їх розиграння та статистичної оцінки результатів моделювання. Такий розиграння здійснюється методом статистичного моделювання (методом Монте-Карло). Отже, статистичне моделювання є самостійним видом моделювання та включається в імітаційне моделювання тільки в разі необхідності моделювання ймовірнісних систем і процесів.

Розробку імітаційної моделі доцільно починати з простої початкової моделі, яку в міру уточнення вхідних даних і характеристик системи ускладнюють та коригують, тобто адаптують до нових умов. Водночас імітаційна модель має залишатися досить наочною, тобто структура її має відповідати структурі модельованої системи.

Рівень деталізації модельованої системи вибирається з урахуванням мети моделювання, обмежень та вихідних даних.

Методологія системного аналізу стосовно імітаційного моделювання включає такі основні етапи:

1. Формулювання проблеми та змістовної постановки завдання.
2. Розробка концептуальної моделі.
3. Розробка та програмна реалізація імітаційної моделі, яка включає:
 - а) вибір мови програмування, якою буде реалізована імітаційна модель;
 - б) розробка структурної схеми імітаційної моделі та опис її функціонування;
 - в) програмна реалізація імітаційної моделі.
4. Перевірка адекватності імітаційної моделі.
5. Планування експериментів на моделі.
6. Оцінка точності результатів моделювання.
7. Інтерпретація результатів моделювання та прийняття рішень.

Перераховані етапи моделювання взаємопов'язані, а сама процедура утворення імітаційної моделі - ітераційна. Це пояснюється тим, що після виконання кожного етапу перевіряється правильність моделі та в разі невідповідності моделі і об'єкта здійснюється повернення до попередніх етапів з метою коригування моделі.

При побудові імітаційної моделі приводиться логіко-математичний опис модельованої системи з урахуванням динаміки її функціонування. Частини моделі, які можливо описати в математичному вигляді, подають як аналітичні залежності. Інші частини моделі можна подати у вигляді докладного словесного опису або алгоритму. Засобом зображення концептуальної моделі може бути діаграма (граф) станів системи.

Імітаційна модель може бути реалізована алгоритмічною мовою програмування загального призначення або з використанням спеціалізованих засобів моделювання. При реалізації складної імітаційної моделі, яка має багато різних компонентів, перевагу належить віддати фаховим засобам моделювання. При використанні алгоритмічних мов з метою уніфікації програмних модулів і принципів структурного та модульного програмування необхідно розробити засоби: управління процесом моделювання, збирання статистичних даних, генерації різних законів розподілів, діагностики помилок при виконанні моделюючої програми. Ці самі засоби є в будь-якій мові мо-

делювання або ІІІІ, орієнтованому на моделювання.

Що до принципів побудови імітаційної моделі доцільно вибирати мову моделювання, орієнтовану на процеси (2). Для цих мов характерна наявність деяких динамічних об'єктів (повідомлень, транзактів), які рухаються по моделі за визначеною схемою. Програма процесу описує шляхи проходження повідомлень у системі. В міру свого просування повідомлення "споживають" деякі ресурси, які також називають активностями. Якщо ресурс зайнятий одним повідомленням, а потрібен у даний момент часу для іншого, то до нього виникає черга.

У різні моменти модельного часу повідомлення можуть виконувати різні оператори моделюючої програми. Досягається це за рахунок квазіпаралельної роботи програми в системному часі, тобто годинник модельного часу не переводитиметься на нове значення системного часу доти, поки не будуть оброблені всі події для даного моменту модельного часу.

Функції управління процесами та їх синхронізацію виконує програма управління моделюванням, яка веде хронологічно упорядкований список подій.

Якщо програмна реалізація імітаційної моделі здійснюється мовою, орієнтованою на процеси, то структуру моделі можна зобразити у вигляді ресурсів та шляхів проходження потоків повідомлень через ці ресурси. Докладніше структурна схема може бути побудована, якщо виділити всі типи подій для потоків повідомлень та дій, що належать до ресурсів.

Діаграма подій служить ефективним проміжним етапом перетворення концептуальної моделі в програму імітаційної моделі і дає змогу в багатьох випадках пропускати етап алгоритмізації моделі та переходити безпосередньо до розробки програми.

Сучасні мови моделювання, орієнтовані на процеси, мають важливу властивість - засіб структурованого описування класів процесів, що дає змогу скоротити розмір імітаційної моделі.

Дослідника, в першу чергу, цікавить функціональна корисність моделі (впевненість у тому, що модель поводить себе справді як реальна система) і переконання в тому, що експерименти з моделлю справедливі та коректні.

Для оцінки адекватності імітаційної моделі необхідно розглянути внутрішню спроможність моделі, тобто її відповідність моделюваної системи і правильність інтерпретації здобутих з допомо-

гою її результатів. Важливим питанням є питання про міру відповідності моделі та системи.

Оцінюючи адекватність моделі, необхідно розглядати весь ланцюг її утворення, починаючи із вхідних даних, формалізованого подання і закінчуючи результатами моделювання. Тому оцінку адекватності розглядають звичайно на концептуальному рівні, рівні реалізації імітаційної моделі і рівні проведення прогонів (розрахунків) моделі.

Основна мета планування експериментів - найбільш глибоке вивчення поведінки модельованої системи при найменших затратах. Для цього складається план експериментів. Використовуються такі типи експериментів: порівняння середніх та дисперсій різних альтернатив; визначення важливості врахування або значення впливу змінних та обмежень, накладених на ці змінні; пошук оптимальних значень на деякій множині можливих значень змінних.

Оцінка точності результатів моделювання залежить від необхідного обсягу вибірки та числа реалізацій. Досягнення необхідної точності результатів моделювання може призвести до істотного збільшення обсягу вибірки, що пов'язано зі значними витратами машинного часу. Тому доцільно застосовувати спеціальні методи зниження дисперсії рішень при обмеженому обсязі вибірки.

Обрані програмні засоби визначають можливі методи проектування імітаційної моделі. Однак незалежно від вибраних засобів для побудови програмних реалізацій імітаційної моделі існують загальні підходи до методів проектування імітаційних моделей.

Найпростішим є варіантний метод. Додержуючись його, проектувальник, спираючись на свій досвід та інтуїцію, послідовно крок за кроком проектує імітаційну модель. У ході розробки для кожної частини модельованої системи розглядається кілька варіантів її відображення в імітаційну підмодель. Це так звана послідовна схема проектування, застосування якої рідко приводить до допустимих проектних рішень і не відповідає загальній схемі системного аналізу для імітаційного моделювання. Така схема передбачає ітераційну процедуру повернення до вже розроблених частин моделі.

Отже ітераційне проектування полягає в тому, що спроектована спочатку імітаційна модель перетворюється в допустиму шляхом ітераційних змін. Таке проектування є за суттю методом "проб та помилок", який через послідовні циклічні зміни приводить до моделі, що задовольняє вимоги точності та адекватності.

Способи внесення змін у модель базуються на принципі направленого дослідження. Щоб застосувати його, можна побудувати гіперповерхню для індексів точності моделі в просторі параметрів імітаційної моделі та оптимізації або хоча б поліпшення показників точності моделі. Сама ж процедура внесення змін у варіант моделі звичайно застосовує перевірку гіпотез. Гіпотези формулюють, виходячи з результатів проектування для попередніх моделей.

Якщо результати порівняння моделі і системи незадовільні, то перш ніж вносити зміни в модель, необхідно сформулювати ряд гіпотез, які мають визначити, що є причиною цієї невідповідності. Гіпотези доцільно формулювати для таких рівнів подання імітаційної моделі: описування структури, алгоритмів поведінки, параметрів та вхідних даних. Пошук причин невідповідності необхідно починати з рівня вхідних даних, для чого оцінюють чутливість моделі до їх зміни. Якщо виявлена значна залежність зміни вихідних даних від незначної зміни вхідних, то необхідно уточнити вхідні дані для моделі і/або локалізувати блоки моделі, на які найбільше впливають ці вхідні дані. Виявлення причин такого сильного зв'язку може вимагати внесення змін в імітаційну модель на рівні структури заміною окремих блоків моделі на докладніші, що, в свою чергу, спричинить зміну внутрішніх параметрів моделі та алгоритмів поведінки.

Параметричне налагодження імітаційної моделі вимагає пошуку найкращих або оптимальних параметрів, за яких ступінь невідповідності між моделлю та системою буде мінімальним. Це типове завдання оптимізації параметрів моделі.

Алгоритми поведінки моделі можуть змінюватись локально для окремих блоків моделі або для моделі в цілому. Такі зміни вимагають більш докладного вивчення поведінки модельованої системи і можуть змінити рівень деталізації в моделі.

Внесення змін у модель на рівні подання її структури - найскладніше налагодження моделі, бо ці зміни можуть спричинити зміни алгоритмів поведінки, параметрів та вхідних даних моделі. Таку перебудову моделі можна починати тільки тоді, коли всі інші можливості вичерпані. Вона може призвести до глобальних змін імітаційної моделі та її заміни новою. Тому перш ніж розпочинати зміни в моделі, необхідно упорядкувати всі гіпотези щодо розміру затрат, які потрібні для зміни моделі. Починати перевірку гіпотез належить з гіпотези, яка вимагає мінімальних затрат, а отже, і

мінімальних змін для імітаційної моделі.

Незалежно від використання варіантного або ітераційного методу існують два принципово відмінних підходи до проектування імітаційних моделей. Перший здійснює таке проектування за схемою згори вниз так зване ієрархічне, багаторівневе або низхідне проектування, другий зсередини догори або висхідне проектування.

Як при низхідному, так і при висхідному проектуванні з метою зменшення розмірності імітаційної моделі однотипні блоки можуть об'єднуватися в класи, причому кожен блок певного класу може мати свій алгоритм поведінки, відмінний від інших елементів. Найбільш ефективну процедуру об'єднання елементів у класи, можна дістати використовуючи низхідний метод проектування імітаційних моделей.

Отже, з точки зору системного аналізу найбільш ефективним методом проектування імітаційних моделей буде такий, який поєднує в собі у певній пропорції низхідне та ітераційне проектування.

Розробка інструментальних засобів для проектування імітаційних моделей з використанням ЕОМ привела до утворення автоматизованих систем проектування, в яких використовується метод інтерактивного, людино-машинного проектування.

Головне завдання інтерактивної системи проектування - організація спільних дій проектувальника та ЕОМ щодо вибору і аналізу варіантів утворення імітаційної моделі шляхом інтеграції їх в єдиний людино-машинний процес вироблення рішень. В таких системах концепція проектування, яка здійснюється обчислювальною машиною, замінюється концепцією проектування з допомогою ЕОМ.

Для утворення програмних реалізацій імітаційних моделей можуть служити готові приватні моделі або підмоделі; конструктори імітаційних моделей, які дають змогу збирати модель з підмоделей; генератори імітаційних програм, які можуть створювати програмні реалізації імітаційних моделей без використання мов програмування користувачем системи. Однак ступінь автоматизації функцій проектування імітаційних моделей у цих системах все-таки не достатній. Підвищити його можливо за рахунок використання інтелектуальних систем проектування.

Розроблена автором ІСІМ орієнтована на моделювання дискретних стохастичних мереж, а її програмний генератор - на мову GPSS.

Створення автоматизованої методики синтезу програмних реалізацій імітаційних моделей визначеного класу вимагає побудови формалізованої моделі у вигляді теоретико-множинного опису.

Ширший клас структур концептуальних та імітаційних моделей можливо подати у вигляді орієнтованих графів виду:

$$G = \langle V, P, Q \rangle, \quad (6)$$

де V, P - скінчена множина відповідно вершин і ребер; Q - функція, яка ставить у відповідність до кожного ребра з P упорядковану пару вершин з V .

Розглядаємо граfi з обмеженнями на ступінь вершини графа. Нехай l_i - ступінь i -ї вершини графа (кількість інцидентних i -ї вершини ребер), l_{i+} - число таких ребер, l_{i-} - число вихідних ребер. Тоді

$$\min l_i = 1; \max l_i = 4; \max l_{i+} = 2; \max l_{i-} = 2. \quad (7)$$

Для програмного генератора вершинам графа відповідає кінцева множина можливих атомарних підмоделей A класів об'єктів K . Характеристика класу K задається сукупністю атрибутів b , які подібні функціям f , визначеним на полі класу. Вони ставлять у відповідність з елементам класу значення, які з точки зору традиційного поняття функції можуть бути однакові. Атрибут визначається як пара $b = \langle i, f \rangle$, де i - ім'я атрибута, яке ідентифікує вершини графа, що належать даному класу; f - функція, визначена на полі класу, яка визначає порядок перегляду вершин графа.

Отже, для стохастичних мереж множина вершин V відображується кількістю атомарних підмоделей A :

$$A = K_1 \cup K_2,$$

де K_1, K_2 - класи об'єктів, модельованих відповідно як одно- та багатоканальні СМО.

Дуги графа стохастичної мережі визначають можливі напрямки передачі повідомлень (транзактів) з підмоделі в підмодель.

Множина A дає змогу створювати узагальнену імітаційну модель за рахунок засобів агрегованого опису класів однотипних моделей.

Множина P визначає створення та реорганізацію зв'язків між підмоделями. Тоді пари

$$\langle a_i, p_j \rangle \mid a_i \in A, p_j \in P, i=1..n, j=1..m,$$

де n, m - кількість відповідно можливих підмоделей і зв'язків між ними, визначають підмодель СМО та можливі напрямки передачі повідомлень з неї.

Нехай F - функція утворення множини таких пар для абстрактної імітаційної моделі, причому $F = Q^{-1}$ та $F = \{f\}$.

Щоб утворити концептуальну модель, досить використати структуру виду

$M1 - \langle A, P \rangle,$

яка вимагає уточнення при утворенні логічної моделі об'єкта. В цьому разі структура $M1$ перетворюється на структуру виду

$M2 - \langle A, P, F \rangle,$

яка дає змогу виконати наступний ланцюг: атомарні підмоделі - правила вибору - параметричне налагодження.

Якщо для концептуального рівня досить застосувати структуру $M1$, визначивши її матрицею можливих переходів, то для логічного рівня використання структури $M2$ є явно недостатнім і поверхневим. Необхідно провести конкретизацію з урахуванням того, що генератор програм орієнтований на конкретну мову моделювання GPSS. Покажемо, як структуру $M2$ можна перетворити на структуру $M3$, що відображає програмну реалізацію імітаційної моделі.

Для цього визначимо специфікацію дискретної системи як структуру:

$M3 - \langle X, Y, S, F1, F2 \rangle,$

де X, Y, S - множина відповідно зовнішніх, внутрішніх і вихідних процесів (алгоритмів передачі повідомлень); $F1, F2$ - відповідно функції переходів і вихідні.

Тоді $A = \{ \langle x, y \rangle \mid x \in X, y \in Y \}$,

де $x, y \in K1 \cup K2$; $F = \langle F1, F2 \rangle$; $P \in S$.

Тепер наведемо зв'язок структури $M3$ з конкретними блоками мови моделювання GPSS:

X - множина блоків GENERATE, TERMINATE;

Y - блоки, пов'язані з визначенням приладів, нагромаджувачів, черг, блоки збирання статистики;

S - блоки, які визначають і/або змінюють маршрут руху транзакту в моделі (TRANSFER, TEST, LOOP та тощо);

$F1$ - блоки, пов'язані з використанням логічних ключів (GATE, LOGICAL);

$F2$ - функції, які визначають дисципліни надходження, обслуговування, руху та виходу транзактів у моделі.

При цьому множину $F2$ можна ще інтерпретувати як таке відображення: нехай a_1 - поточний процес моделі, $\{ a_2 \}$ - множина процесів моделі, куди може бути переданий транзакт з процесу a_1 (при цьому можливе таке включення $a_1 \in \{ a_2 \}$), то

$F2: a_1 \rightarrow \{ a_2 \}.$

Для програмного рівня формалізації модель визначається граматикою мови моделювання GPSS. Цей рівень в процесі автоматизації

програмної реалізації моделі звичайному користувачеві недоступний, за винятком користувача-програміста, тому немає потреби в його формальному описі.

В рамках наведеного формального опису процесу проектування імітаційних моделей необхідно забезпечити ефективний діалог. Такий діалог здійснюється мовою інтерактивної взаємодії з користувачем, яку реалізують через ієрархічну систему меню.

Отже, формальна модель опису об'єкта для проектування його програмної реалізації імітаційної моделі зазнає послідовних змін структур M_1 , M_2 , M_3 , причому, якщо структуру M_1 задає користувач у інтерактивному режимі, відповідаючи на запити системи, то структуру M_3 має створювати програмний генератор, використовуючи лінгвістичний процесор (ЛП) та базу даних моделей.

Основне призначення ЛП полягає в тому, що він шляхом інтерактивної взаємодії з користувачем деякою мовою L має побудувати підмножину таких висловлень D з деякої множини B , що тільки висловлень D завжди істинне. Отже, ЛП задається четвіркою $\langle L, D, B, T \rangle$, де множина T визначає деяку теорію відносно висловлень B . З прагматичної точки зору множина B визначає всі можливі відповіді на запитання з меню, які утворюють мову L . Підмножина D - тільки ті відповіді, за якими алгоритм роботи ЛП, що визначає теорію T , буде семантично та синтактично правильні оператори GPSS програми.

Мова L - це мова інтерактивного проектування моделей, яка утворюється із сукупності меню. Множина B відповідає базі даних імітаційних підмоделей і правилам руху транзактів по моделі, які закладені в ЛП, а підмножина висловлень D визначає параметричні налагоджені імітаційні моделі для конкретної реалізації.

ЛП - це багаторівнева програма, реалізована мовою Paskal, в якій активізуються підпрограми, процедури та функції при генерації деякої імітаційної моделі, та здійснюється її послідовне проектування. Сформовані з допомогою ЛП моделі записуються в бібліотеку моделей (в каталог користувача) для імітаційного моделювання і надходять на вхід підсистеми проведення експериментів і статистичної обробки результатів моделювання (ПЕСО), в режимі інтерпретації GPSS- програми.

Структура багатовіконного меню включає головне меню та підменю (рис. 2). Мова взаємодії з користувачем російська.

Структура M_1 в ICIM визначає вид стохастичної мережі, яка вводиться в режимах утворення нової структури або модифікації за

допомогою матриці переходів між її вершинами.

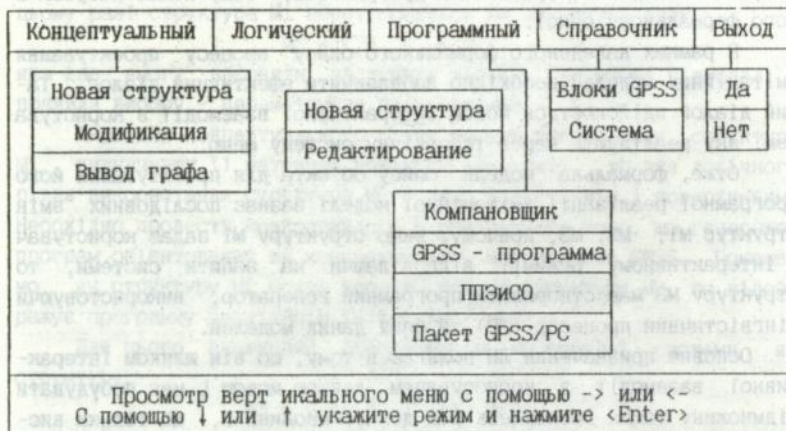


Рис. 2

Уводиться жорстка типізація (декларативний спосіб опису) вершин стохастичної мережі. Користувач може задавати або не задавати потік повідомлень для вершини, вибирати вид СМО та вводити параметри потоку та СМО. Це здійснюється в режимі "Логический" з головного меню (рис. 3,4), який має режими для введення нової структури та для редагування введеної. Таким чином на логічному рівні модель визначається у вигляді структури M1 і частково M2.

На програмному рівні задається структура M3. Цей режим має такі можливості:

задавання правил (алгоритмів) руху транзактів в моделі, якщо вони встановлюються, зміну параметрів транзактів (атрибутів повідомлень) і виходи з підмоделі (рис. 5). Передбачені виходи із звичайною прямою передачею в іншу підмодель, з багатократним дообслуговуванням, альтернативний вихід за деякої умови;

опис умов проведення експериментів зі збиранням статистичних даних у таблиці (рис. 6);

компування GPSS-програми;

виведення тексту програми з можливістю його редагування;

запуск пакету GPSS/PC з створеною програмою;

одержування довідок про роботу системи і блоком GPSS.

вий характер. Тому, будуючи параметричні моделі, для вихідних даних імітаційної моделі доцільно обмежитись тільки моделями авторегресії. Авторегресійний процес порядку k зі сталим коефіцієнтами визначається рівнянням

$$Y_t = a_0 + a_1 Y_{t-1} + a_2 Y_{t-2} + \dots + a_k Y_{t-k} + e_t,$$

тобто значення ряду в момент t виражається через попередні значення та випадкове збурення e в момент t .

ПЕРЕДАЧА СООБЩЕНИЙ	
Число вершин: 10	Тип сети: замкнутая
Рассматриваемая вершина: 2	Сообщения передаются в 3 4
Условия передачи в 3 : по вероятности Задерживать сообщения: Y	
значения в долях 1000: 400	Закон распределения для вре-
Кратность обслуживания: многократная	мени задержки: равномерный
Прекращение обслуживания: по параметру	Параметры: первый: 10
значение: 1	второй: 5
Изменять параметры сообщений: Y	
сколько параметров: 1	
номер параметра сообщения: 3	
значение возрастает на сколько единиц: 1	
F10 - просмотр концептуального графа, ENTER: для возврата	
↑↓: выбор режима ПРОБЕЛ: выбор альтернативы ENTER: ввод ESC: выход	

Рис. 5

УСЛОВИЯ ЭКСПЕРИМЕНТА	
Число вершин: 10	Тип сети: замкнутая
Закончить моделирование: по времени	Проверять модель на циклы
значение: 400000	регенерации: Y
Уничтожать сообщения: N	номер начальной вершины: 1
В скольких вершинах:	номер конечной вершины: 10
номера вершин:	
Выдавать статистику о времени	
прохождения сообщений: Y	
номер начальной вершины: 1	
номер конечной вершины: 10	
F10 - просмотр концептуального графа, ENTER: для возврата	
↑↓: выбор режима ПРОБЕЛ: выбор альтернативы ENTER: ввод ESC: выход	

Рис. 6

На практиці звичайно використовуються авторегресійні моделі

першого та другого порядку, що відповідає випадковим процесам Маркова та Юла. Параметри a_1, a_2, \dots, a_k оцінюються через коефіцієнти автокореляції.

Для утворення параметричних моделей використана методика Бокса-Дженкінса. Аналіз результатів показав, що авторегресійна модель може використовуватися тільки після відкидання даних перехідного режиму. В усіх випадках не доводилось використовувати авторегресійні моделі більш, ніж другого порядку.

Оскільки більшість імітаційних експериментів проводиться з метою оцінки математичного сподівання деякої вихідної величини з заданою точністю, то використання авторегресійної моделі дає змогу визначити тенденцію зміни досліджуваної вихідної величини в часі та прогнозувати її подальшу зміну в майбутньому. Причому, якщо вихідні дані одного прогону імітаційної моделі є реалізацією деякого випадкового процесу, то завдяки використанню параметричної моделі можна передбачати середню оцінку для прогнозованого інтервалу часу.

У момент часу $t+1$ модель авторегресії може бути подана як

$$y_{t+1} - a_1 y_t - a_2 y_{t-1} - \dots - a_k y_{t+1-k} = \epsilon_{t+1}. \quad (8)$$

Якщо взяти з обох частин умовне математичне сподівання при фіксованих до моменту t минулих значеннях, отримуємо

$$y_{t+1} - a_1 y_t - a_2 y_{t-1} - \dots - a_k y_{t+1-k} = 0.$$

Рівняння має вирішення

$$y_{t+1} = b_0^{(t)} f_0[t+1] + b_1^{(t)} f_1[t+1] + \dots + b_k^{(t)} f_k[t+1], \quad l > 0,$$

де $f_0[t+1], f_1[t+1], \dots, f_k[t+1]$ - функція усереднення до часу l .

Для заданого t $b_j^{(t)}$ стали для всіх l . Зокрема при $k=1$ для моделі авторегресії першого порядку маємо

$$\hat{y}_{t+1} = \hat{y}_t a^1,$$

де $\hat{y}_t = y_t - \mu$.

Для стаціонарного процесу слід виконувати умову $-1 < a < 1$. При $l \rightarrow \infty$ \hat{y}_{t+1} буде прямувати до μ , тобто $\hat{y}_{t+1} - \mu \approx 0$.

Таким чином, при великих значеннях випередження l з мінімальною середньоквадратичною похибкою значення ряду повинно експоненціально прямувати до його математичного сподівання. Аналогічно для процесу авторегресії другого порядку.

Звичайно, дуже складним є визначення інтервалу випереджування l , оскільки для цього не існує точних методів. У разі використання авторегресійної моделі прогноз збігається до деякого сталого значення, якщо наступне значення y_t дорівнює попередньому y_{t-1}

або для моделі першого порядку

$$y_t = y_{t-1} + a_0 / (1 - a_1). \quad (9)$$

Звідси зрозуміло, що число точок прогнозу до моменту збіжності до сталого значення визначатиметься тим, на скільки останнє значення рівня початкового ряду відрізнятиметься від величини, розрахованої за формулою (9).

Оскільки початкове значення $a_1 = r_1$, тобто величині оцінки коефіцієнта автокореляції первісного ряду з лагом 1, то число точок передбачення до моменту, коли $y_t - y_{t-1}$, буде великим, якщо значення часового ряду дуже корельовані. В цьому разі необхідно керуватися правилом: число точок прогнозу має бути не більшим за довжину первісного ряду. В решті випадків прогноз здійснюється доти, поки не буде виконана умова (9).

Для моделювання авторегресії другого порядку умова (9) буде матиме вигляд $y_{t-2} - y_{t-1} - y_t = a_0 / (1 - a_1 - a_2)$.

Ураховуючи те, що авторегресійна модель часового ряду містить інформацію про взаємозв'язок значень змінної у часі, а функція розподілу - інформацію про ймовірність набуття випадковою величиною визначеного значення, бажано було б відтворити можливі значення для прогнозованих рівнів вихідної змінної імітаційної моделі. Це можна зробити для стаціонарних у вузькому розумінні слова процесів, у яких функція розподілу не змінюється у часі, з допомогою методу статистичних іспитів.

Тоді алгоритм використання параметричної моделі буде такий.

1. Побудова прогнозованих значень за допомогою авторегресійної моделі (8).

2. Розіграш функції розподілу, отриманої за значеннями первісного ряду, для кожного значення прогнозу з урахуванням зміщення середнього значення первісного ряду та здобутого значення для точки прогнозу. Оскільки врахування зміщення середнього може привести до значень, які не можуть бути здобуті на імітаційній моделі, то в разі перевищення максимально можливої величини треба запам'ятати максимальне значення. Аналогічно, якщо значення менше за мінімально можливе, треба запам'ятати мінімальне значення.

3. Додавання розіграних даних до первісного ряду. Оскільки дані, які додають, не корельовані, то значення автокореляційної функції зменшаться тим більше, чим більше прогнозованих даних буде додано до первісного часового ряду.

4. Оцінка точності результатів. Через те, що процедура утво-

рення авторегресійної моделі вимагає розрахунків автокореляційної функції, то для оцінки точності результатів моделювання доцільно застосовувати метод розкладання всієї вибірки, яка включає первісні значення та значення, які прогнозують, на підвибірки. Обсяг підвибірки визначається за автокореляційною функцією, коли коефіцієнти автокореляції стають незначущі. Середні значення підвибірок будуть незалежними і їх можна використати для утворення довірчого інтервалу. Якщо отримана точність виявиться недостатньою, слід збільшити обсяг початкової вибірки для утворення параметричної моделі і повторити процедуру передбачення.

Така методика дає змогу побудувати адаптивний алгоритм для класу регенеруючих моделей. В цьому разі збирання даних з імітаційної моделі закінчується із припиненням циклу регенерації моделі. Потім будується параметрична модель, здійснюється прогнозування та оцінюється точність вихідної змінної. Якщо одержана точність виявляється недостатньою, то імітаційний експеримент поновлюється з нового циклу регенерації. Здобуті дані додають до вже існуючих і знову оцінюють точність. У разі потреби повертаються на перший етап побудови авторегресійної моделі.

Аналіз часу перебування заявки Trg в СМО виду M/M/1, який одержано при прогоні імітаційної моделі з 120 циклами регенерації та коефіцієнтом завантаження приладу 0,9, дав оцінку середнього значення часу Trg у системі $2031,61 \pm 717,2153$ з довірчою ймовірністю 0,9. Оцінка середнього значення Trg за отриманим рядом зі СМО і відтвореними можливими значеннями для прогнозованих рівнів становить 1821,92. Аналітичні розрахунки дають значення середнього часу 1800. Щоб дістати таке середнє значення з 5% відхиленням від середнього та довірчою ймовірністю 0,9, на імітаційній моделі слід промодельовати близько 12000 циклів регенерації, при цьому початковий обсяг вибірки зростає більш ніж у сто разів.

Для проведення експериментів і статистичної обробки результатів моделювання розроблена підсистема ПЕСО, основне завдання якої полягає у збиранні даних за результатами прогону імітаційної моделі, статистичній обробці їх і подання результатів моделювання. З метою прискорення процесів моделювання та проведення поточної декомпозиції в імітаційній моделі у ПЕСО включені засоби утворення параметричних моделей часових рядів з їх прогнозуванням і відбудовуванням реалізацій випадкових процесів, а також засоби реорганізації структури GPSS-моделі для зменшення її розмірності.

У правій частині рис. 7 показано відновлений ряд з допомогою моделі (8). Первісний ряд з 386 значеннями отримано від СМО.

(X 1000)

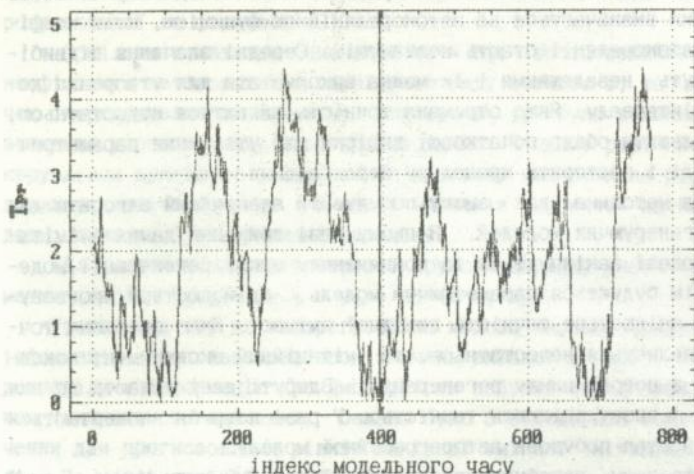


Рис. 7

Для побудови параметричної моделі часового ряду першого або другого порядку досить знати один або два останніх члена ряду та функцію розподілу помилки, що робить процедуру генерації дуже простою та швидкою. Це дає змогу використовувати її для зменшення розміру імітаційної моделі шляхом заміни ряду блоків моделі на аналітико-статистичні моделі часових послідовностей. Одна з них використовується для визначення моментів модельного часу настання подій, а інші - для вихідних змінних, які отримано з імітаційної моделі для цих подій.

У ІСІМ структура імітаційної моделі задається у вигляді графа (6) з обмеженнями виду (7). Для графа J такої структури можна здійснити потокове агрегування виду

$$\{J^b | l_{p+} = 1, l_{k-} = 1\} \in J,$$

де l_{p+}, l_{k-} - число вхідних ребер відповідно в підграфі та з нього.

З множин таких підграфів вибирається підграф J^b з інтенсивністю потоку заявок λ^b .

$$\lambda^b = \max(n^b) / t | \Gamma^b \rightarrow 1,$$

де n^b - кількість заявок, які пройшли через підграф b ; t - час прогону моделі (час моделювання); Γ^b - значення автокореляційної функції в лагом 1 для потоку заявок з інтенсивністю λ^b .

Цей підграф замінюється на мультиграф $J^m | n+ - 1, k - 1$ з однією мультивершиною i з однією вхідною та вихідною дугою. На програмному рівні така операція здійснюється заміною послідовності блоків GPSS-програми на один блок затримки ADVANCE з функцією розподілу, отриманою за часовим рядом або з використанням авто-регресійної моделі з відбудовуванням.

Меню для підсистеми ПЕСО наведено на рис. 8. При генерація часового ряду, користувач вводить ім'я файла GPSS-звіту та числа даних перехідного процесу, які він хоче відкинути. Для генерації функції розподілу, він задає ім'я файла GPSS-програми, куди записуватиметься побудована функція розподілу. Остання може будуватися з таблиці, яку дістали в результаті прогону GPSS моделі або з таблиць, одержаних за часовим рядом з допомогою пакету STATGRAPHICS.

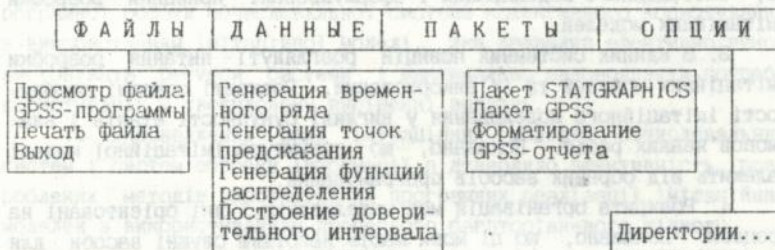


Рис. 8

У роботі розроблена автоматизована процедура визначення найкращих рішень щодо управління та удосконалення технологічної ділянки. Вона ітераційна, застосовує перевірку гіпотез і пов'язана з внесенням змін у технологію обробки деталей на дільниці. З цієї метою виконуються такі етапи:

1. Виявлення причин зниження продуктивності дільниці та зменшення доходу від робіт, що виконуються.
2. Бисунення гіпотез та попередній аналіз їх розумності.

3. Перевірка гіпотез та порівняння здобутих результатів і упорядкування їх за зменшенням витрат.

4. Видача рекомендацій щодо вдосконалення технологічної дільниці.

Приклади імітаційних систем і моделей для обробки, передачі даних, які побудовані по розробленим принципам і методам, наведені в роботі. Зокрема імітаційна модель галузевої мережі підготовки, передачі та обробки інформації впроваджена в промислову експлуатацію. Ці моделі також використались для аналізу роботи різних обчислювальних центрів.

ОСНОВНІ РЕЗУЛЬТАТИ І ВИСНОВКИ

1. Аналіз динамічних систем дав змогу розробити найбільш загальну модель таких систем у вигляді схеми процесів. Ця модель дозволяє побудувати моделюючі алгоритми, здійснювати проектування генераторів імітаційних моделей, описувати взаємодію користувача в інтерактивній системі.

2. Виділені основні особливості математичного, статистичного, імітаційного моделювання і сформульовані принципи розробки імітаційних моделей.

3. З єдиних системних позицій розглянуті питання розробки імітаційних моделей та їх використання. Розкриті основні особливості імітаційного моделювання у вигляді сукупності етапів і взаємопов'язаних робіт. Показано, що структура імітаційної моделі залежить від обраних засобів програмування.

4. Розкриті організації мов моделювання, які орієнтовані на процеси. Показано, що ці мови мають найбільш слушні засоби для проектування складних імітаційних моделей з використанням принципів низхідного і об'єктного проектування імітаційних програм.

5. Подання структури імітаційної моделі у вигляді графа стохастичної мережі з послідуємим перетворенням у модель процесів дозволило розробити єдиний підхід до проектування ІСІМ. Користувачу надаються можливості агрегованого опису концептуальної та імітаційної моделей за рахунок виділення класів об'єктів та їх параметричного налагодження на конкретну реалізацію моделі.

6. Створена компактна ІСІМ, яка працює з великою швидкістю і використовує генератор програм з лінгвістичним процесором. ІСІМ дозволяє користувачу не програмісту будувати програми імітаційних

моделей мовою GPSS для класу дискретних стохастичних мереж. Відміні особливості даної системи полягають у тому, що користувач працює у багатовіконному меню з виділеними класами об'єктів, наявність засобів потокового агрегування та прискореного моделювання, організація взаємодії з пакетом програм статистичної обробки інформації.

7. Для прискорення моделювання повільних, з незмінними ймовірнісними характеристиками процесів пропонується використовувати авторегресійну модель часового ряду з відновленими значеннями для прогнозованих рівнів вихідних змінних імітаційної моделі. Дослідження різних СМО показують, що використання авторегресійних моделей дозволяє скоротити обсяг вибірки від двох до ста разів у залежності від ступеня корельованості даних.

8. Для автоматизації процедури прийняття рішень з використанням імітаційної моделі запропонована методика на основі формулювання гіпотез і вибору найкращої з них за критерієм мінімізації витрат на їх реалізацію, яка ілюструється на прикладі імітаційної моделі виробничої ділянки.

9. Запропоновано алгоритм рішення задачі планування мультипрограмної роботи обчислювальної системи колективного користування з використанням імітаційної моделі, яка дозволяє ефективно використовувати ресурси системи і максимально задовольняти потреби користувачів у своєчасному вирішенні завдань.

10. Впровадження ряду імітаційних моделей обчислювальних систем і систем обробки інформації підтвердило ефективність розроблених методів проектування програмних реалізацій імітаційних моделей з використанням процесного багаторівневого підходу.

ПЕРЕЛІК ОСНОВНИХ НАУКОВИХ ПРАЦЬ

1. Основы системного анализа и проектирования АСУ: Учеб. пособие /А.А. Павлов, С.Н. Гриша, В.Н. Томашевский и др.; Под общ. ред. А.А. Павлова. - К.: Выща шк., 1991. -367 с.

Автором написана глава 4, в якій з позиції системного аналізу розкрито методологія імітаційного моделювання в вигляді сукупності етапів і робіт. Показано, що структурна схема імітаційної моделі залежить від обраних засобів програмування.

2. Томашевський В.М. Імітаційне моделювання систем і процесів.: Навч. посібник.- К.: ІСДО, 1994. -124 с.

Із системних позицій розкрита суть імітаційного моделювання, його внутрішній, прихований від користувача механізм. На базі системи моделювання GPSS/PC описується інтерактивна система, до якої включені засоби прискореного моделювання повільних процесів і потокової декомпозиції швидких процесів.

3. Антонюк А.И., Томашевский В.Н. Системный подход к проектированию программных реализаций имитационных моделей // Информатизация процессов экономического развития Украины: Сб. науч. тр. - Киев: ГНИИМЭ Минэкономики Украины, 1994. - С. 159-165.

Використан системний підхід для проектування програмних реалізацій імітаційних моделей.

4. Купрашвили А.Ю., Савустьяненко Э.И., Томашевский В.Н. Организация интерактивной системы моделирования // Электрон. моделирование. - 1987. - № 1. - С. 16-19.

Запропоновано багаторівневий підхід побудови інтерактивної системи імітаційних моделей з параметричною базою даних і лінгвістичним процесором для мови моделювання GPSS.

5. Зайченко Ю.П., Томашевский В.Н., Гнатовский В.Н. Имитационная модель ВЦКП с развитой абонетской сетью. УСИМ, № 2, 1980. - С. 8-13.

Розроблена імітаційна модель мережі обчислювальних центрів колективного користування.

6. Томашевский В.Н. Имитационная модель вычислительных центров коллективного пользования. В сб. Вестник КПИ, серия техническая кибернетика, вып. 4, 1980. - С. 78-80.

Розроблена імітаційна модель мережі обчислювальних центрів колективного користування.

7. Томашевский В.Н., Гнатовский В.Н. Имитационная модель ВЦКП. Киев, Украинский республиканский фонд алгоритмов и программ, 1980. - 131 с.

Розроблена програмна реалізація імітаційної моделі мережі обчислювальних центрів колективного користування мовою АЛСИМ.

8. Вишталь Д.М., Томашевский В.Н. Об одном алгоритме планирования работы вычислительной системы ЕС ЭВМ. В сб. Вестник КПИ, серия техническая кибернетика, вып. 8, 1984. - С. 40-44.

Запропоновано аналітико-імітаційний підхід для планування робіт обчислювальної системи ЕС ЕОМ.

9. Томашевский В.Н., Вервицкая Н.В. Интеллектуальная система моделирования процессов, протекающих с разными скоростями. В сб.

І-а Української конференція з автоматичного керування Автоматика-94.: Тез. доп. - Київ: Ін-т кібернетики АН України, 1994, ч. 1.- С. 215.

Запропонована процесна схема ІСІМ, до якої включені засоби прискореного моделювання повільних процесів.

10. Томашевський В.М. Використання системного підходу при проектуванні програмного генератора імітаційних моделей. / 4-а Міжнародна науково-практ. конф. "Україномовне програмне забезпечення УКРСОФТ - 94", Львів, 1994.- С.

11. Томашевский В.Н. Интерактивная система имитационного моделирования стохастических сетей. /12-я Белорусская Международная Зимняя школа-семинар "Исследование систем и сетей массового обслуживания BWWQT-96", Гродно, 1996.- С.

12. Обушев И.Л., Печурин Н.К., Томашевский В.Н. Моделирование отраслевой сети ВЦ Минэлектротехпрома. В сб. Тезисы докладов секции НТС Минэлектротехпрома, Таллин, 1982.- С. 39.

Запропоновано імітаційно-аналітичний підхід для моделювання мережі для галузі.

13. Зайченко Ю.П., Томашевский В.Н. Имитационная модель функционирования автоматизированной диспетчерской службы в режиме оперативного управления сетью ВЦ. В сб. Материалы всесоюзной конференции "Математическое обеспечение моделирования сложных систем", ч. 2, Киев, 1977.- С. 120-122.

Розроблена імітаційна модель для диспетчерської служби мережі обчислювальних центрів.

14. Зайченко Ю.П., Томашевский В.Н., Вишталъ Д.М. Планирование мультипрограммной работы вычислительной системы. В сб. Тезисы докладов республиканского семинара, Кишинев, 1981.- С. 88-91.

Розроблена задача планування мультипрограмної роботи обчислювальної системи та алгоритм її розв'язання.

15. Томашевский В.Н. Аналитико-имитационная модель для комплекса программ АСУ-ГИВЦ (производство). В сб. Тезисы докладов секции НТС Минэлектротехпрома, Таллин, 1986.- С. 83-84.

16. Методические указания по дисциплине "Моделирование систем" для специальности "Автоматизированные системы управления"/ Сост. В.Н. Качура, В.Н. Томашевский. - К.: УМК ВО Минвуза УССР, 1988.- 26 с.

Розроблені принципи побудови імітаційних моделей, показана залежність структури імітаційної моделі від мови моделювання.

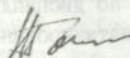
Томашевский В.Н. Методы проектирования интерактивных систем имитационного моделирования. Рукопись. Диссертация на соискание ученой степени доктора технических наук по специальности 05.13.04 - "Автоматизированные системы управления и системы обработки информации". Национальный технический университет Украины "КПИ". Киев, 1996.

Защищается 16 научных работ, в которых рассмотрены проблемы, связанные с разработкой интерактивных систем имитационного моделирования для стохастических сетей. На основе теоретико-множественного описания процессов и лингвистического процессора создана система моделирования ИСИМ для пользователя не программиста, позволяющая проводить потоковое агрегирование для быстрых регенерирующих процессов. ИСИМ имеет высокое быстродействие и требует незначительных ресурсов ПЭВМ. Разработаны процедуры принятия решений по результатам имитационного моделирования.

Tomashevski V.N. Methods for Designing Interactive Simulation Systems. A Manuscript. Tesis for a degree of Doctor's of Technical Sciences, speciality - 05.13.04 - "Automated Control and Data Processing Systems". National Technical University of Ukrain "KPI". Kiev, 1995.

It is defended 16 scientific works. The thesis covers issues related to designing interactive simulation systems for queueing networks. ISIM simulation system for non-programmers, which allows to perform flow aggregation for quick regenerating processes, has been developed on the basis of theoretic set descriptions of processes and a linguistic processor. ISIM is a high-speed system, which requires low resources of a microcomputer. Procedures for decision-making on the simulation results have been developed.

Ключові слова: імітаційне моделювання, моделі процесів, часові ряди, стохастичні процеси, генератори програм, інтерактивні системи.



Підп. до друку 03.01.96. Формат 60×84^{1/16}. Папір
друк. № 2. Друк офсетний. Умовн. друк. арк. 186.
Умовн. фарбо-відб. 1,97. Облік-вид. арк. 20.
Тираж 100. Зам. № 6-10.

Фірма «ВІПОЛ»
252151, Київ, вул. Волинська, 60.

453171

AB 33.917