

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ХАРКІВСЬКА НАЦІОНАЛЬНА АКАДЕМІЯ МІСЬКОГО
ГОСПОДАРСТВА

В.О.ЛЕЛЮК, О.В. ЛЕЛЮК, М.П. ПАН

УДОСКОНАЛЕННЯ БІЗНЕС-СИСТЕМ

НАВЧАЛЬНИЙ ПОСІБНИК

За редакцією В.О.Лелюка

Рекомендовано Міністерством освіти і науки України
для студентів вищих навчальних закладів

Том 2

**Аналіз і удосконалення бізнес-систем
динамічними методами**

Харків – ХНАМГ – 2010

УДК 681.3:51

Лелюк В.О., Лелюк О.В., Пан М.П. Удосконалення бізнес-систем / Під ред. В.О.Лелюка: Навчальний посібник. В 2-х т. Том 2. Аналіз і удосконалення бізнес-систем динамічними методами – Х.: ХНАМГ, 2010. - 121 с.

У томі 2 описано імітаційні моделі, етапи їх створення і використання, проаналізовані базові методології та інструментарій імітаційного моделювання. Розглянуто сучасні пакети програм і практичні приклади їх застосування для аналізу й удосконалення бізнес-систем.

Для студентів спеціальності «Менеджмент організацій», а також викладачів, аспірантів і спеціалістів.

Рецензенти:

**Завідувач кафедри комп'ютерних технологій Української інженерно-педагогічної академії,
проф., д.т.н., академік МАІ А.Т. Ашерів**

**Директор Науково-дослідного центру індустріальних проблем розвитку НАН України,
проф., д.е.н. М.О.Кізим**

Рекомендовано Міністерством освіти і науки України
як навчальний посібник для студентів вищих навчальних закладів,
рішення № 1.4/ Г-65 від 10.01.09

ISBN 978-966-695-142-0

© В.О.Лелюк, О.В.Лелюк,
М.П.Пан, 2010
© ХНАМГ, 2010

ЗМІСТ

	Стор.
Вступ	4
ЧАСТИНА 5 Методологія та інструментарій імітаційного моделювання	
Розділ 12 Моделі і методи імітаційного моделювання	7
12.1 Сутність, об'єкти і області застосування імітаційного моделювання.....	7
12.2 Особливості імітаційного моделювання.....	13
12.3 Основні етапи імітаційного моделювання.....	18
12.4 Моделі і методи структуризації й формалізації імітаційних систем.....	29
12.5 Моделі системної динаміки.....	39
12.6 Імітаційне моделювання в системі ARIS.....	52
Розділ 13 Пакети імітаційного моделювання	54
13.1 Короткий огляд створення й розвитку пакетів.....	54
13.2 Опис пакетів імітаційного моделювання.....	56
ЧАСТИНА 6 Досвід імітаційного моделювання бізнес-систем	
Розділ 14 Досвід застосування методами системної динаміки	65
14.1 Моделювання впливу зміни процентної ставки на сальдо банківського рахунку.....	65
14.2 Моделювання розподілу доходів фірми від продаж й від сервісного обслуговування.....	70
14.3 Моделювання кредитування підприємств.....	77
14.4 Моделювання реалізації інноваційних енергозберігаючих проектів у житлово-комунальному господарстві.....	83
14.5 Моделювання виводу на ринок інноваційних продуктів.....	87
14.6 Моделювання модернізації мереж водопостачання.....	92
Розділ 15 Моделювання надходження заявок на будівництво і їх обслуговування методом СМО	99
15.1 Опис процедури обслуговування забудовників.....	99
15.2 Прогнозування потоку заявок.....	102
15.3 Умови проведення дослідження.....	105
15.4 Дослідження динаміки обслуговування забудовників.....	108
Висновки	115
Список літератури	117

ВСТУП

Том 2 присвячено динамічним методам аналізу і удосконалення бізнес-систем з використанням імітаційного моделювання. Традиційними сферами його застосування є виробничі, транспортні, інформаційні і телекомунікаційні системи [27], бізнес-процеси, логістичні мережі [5, 46], маркетинг [55, 58]. Також його об'єктами є економічні реформи, регіональні процеси і проблеми розвитку міст [11,18,25,36,44], соціальні і політичні проблеми [39], нарешті, і глобальні світові процеси [45].

Імітаційні моделі застосовуються для дослідження систем, коли потрібно виявити і проаналізувати «вузькі місця», динаміку функціонування систем, коли бажано спостерігати хід процесу протягом певного часу. Це дає можливість менеджерам успішно реорганізувати бізнес, ефективно укомплектувати штат підприємства, зробити кращі інвестиції. Вони корисні також для дослідження **стохастичних систем**, тобто систем, на які впливають численні випадкові фактори складної природи. Ці моделі застосовують для проведення досліджень в умовах **невизначеності**, при неповних і неточних даних, для яких використання математичних моделей має обмеження.

В системах підтримки прийняття рішень головна перевага застосування імітаційного моделювання полягає в тому, що аналітик завжди може досліджувати велику кількість варіантів рішень, програти різні сценарії при різних вхідних даних щоб одержати відповіді на питання “Що буде, якщо ...”. Це дає можливість перевіряти нові стратегії і рішення, а також вивчати можливі ситуації. Імітаційна модель дозволяє **прогнозувати** і досліджувати процеси розвитку, коли мова йде про систему, яка проектується, тобто в тих випадках, коли реальної системи ще не існує. В імітаційній моделі можуть бути різні рівні деталізації процесів. Вона створюється **еволюційно**, без різких й істотних змін.

Треба враховувати і **обмеженості імітаційного моделювання**. Його техніка рішення проблем є ітеративною, експериментальною. Вона є комбінацією науки і мистецтва. Аналітикові важко оцінити адекватність моделі і знайти

джерело помилок. При цьому збір, аналіз й інтерпретація результатів вимагає гарного знання теорії імовірності та статистики. Однак дуже часто цей метод є єдиним способом створення моделі і дослідження складної системи, особливо якщо з нею не можна провести експеримент.

Том 2 складається з частин 6, 7, які містять 4 розділи. У **розділі 14** описані моделі, етапи їх створення, базові методи імітаційного моделювання, які використовують системну динаміку. При підготовці цього опису був використаний матеріал перших чотирьох лекцій посібника Н.Н.Личкіної [26], підготовлений для слухачів програми eMBI Академії Ай Ти і Московського державного університету управління. В цьому посібнику дохідливо розкриті поняття і сутність методу, його технологічні етапи, концепції структуризації і формалізації імітаційних систем. Для тих студентів, хто хоче більш глибоко оволодіти цим методом, можна рекомендувати лекції 5-7 цього посібника, де викладені відомості по засобам автоматизації імітаційного моделювання, іспиту та дослідження властивостей моделей і технології постановки та проведення направленою обчислювального експерименту на імітаційній моделі.

Крім традиційних методів імітаційного моделювання, У **розділі 12** описаний також метод, в якому застосовані моделі реальних бізнес-процесів, що розбудовуються в інструментальній системі ARIS.

В **розділі 13** зроблено короткий огляд створення і розвитку сучасних пакетів програм імітаційного моделювання: Process Charter, Powersim, Ithink Extend+BPR, Pilgrim, Vensim. Для їх опису і аналізу був використаний навчальний посібник Ю.О.Кузнецова і В.І.Перової [21], підготовлений в рамках інноваційної освітньої програми підвищення кваліфікації у Нижегородському державному університеті ім. Н.І.Лобачевського.

Розділ 14 присвячено досвіду застосування динамічних методів для оптимізації пропускних здібностей елементів системи, покращення політики цін, умов договорів закупки і продажу товарів. У п.14.1, 14.2 приведені взяті з [48] два класичних приклада моделювання впливу зміни процентної ставки на сальдо банківського рахунку, і розподілу доходів фірми від продаж і від

сервісного обслуговування. Приклад моделювання процесу кредитування підприємств (п.14.3) взято з [7].

Матеріал підрозділів **14.4-14.6**, щодо моделювання реалізації інноваційних енергозберігаючих проектів у житлово-комунальному господарстві, виводу на ринок інноваційних продуктів і модернізації мереж водопостачання, підготовлено співавтором даного навчального посібника М.П.Паном

У **розділі 15** приведено результати дослідження динаміки надходження заявок на будівництво і роботи з ними, виконаного в рамках магістерської роботи О.С.Мордовцева з використанням методу масового обслуговування. Подібне моделювання для цієї теми раніш проводили в своїх магістерських роботах В.В.Гриненко і А.І.Ткаченко

Викладений в посібнику матеріал повинен сприяти набуттю знань і підготовці сучасних менеджерів, здатних працювати в умовах жорстокої конкуренції. Він потрібен не тільки при вивченні системних дисциплін, а і може бути використаний при курсовому проектуванні по дисципліні «Менеджмент організацій» і при виконанні дипломних та магістерських робіт.

Розглянуті методології та інструментарій аналізу і удосконалення бізнес-систем можуть бути корисними також для проведення консалтингових і науково-дослідних робіт. Особливо це торкається динамічних методів моделювання бізнес-процесів і методів їх реінжинірингу за допомогою інструментальної системи **ARIS**.

ЧАСТИНА 5

МЕТОДОЛОГІЯ ТА ІНСТРУМЕНТАРІЙ ІМІТАЦІЙНОГО МОДЕЛЮВАННЯ

Розділ 12

Моделі і методи імітаційного моделювання

12.1 Сутність, об'єкти і області застосування імітаційного моделювання

12.1.1 Загальні відомості про імітаційне моделювання

Створення і дослідження моделей дійсності – це один з основних методів пізнання, що містить з'ясування або відтворення тих або інших властивостей реальних об'єктів, предметів і явищ за допомогою інших об'єктів, процесів, явищ, або за допомогою абстрактного опису у вигляді зображення, плану, карти, сукупності рівнянь, алгоритмів і програм. Модель являє собою абстрактний опис об'єкта, процесу, проблеми, поняття в деякій формі, відмінної від форми їх реального існування.

Моделювання засноване на наявності подібності різноманітних природних і штучних систем або подоби деяких властивостей – геометричних, структурних, функціональних, поведінкових. Виділяють такі основні види моделювання [2]: **фізичне, структурно-функціональне, математичне, концептуальне й імітаційне.** Фізичне моделювання ґрунтується на схожості фізичних явищ і, як слідство, на співвідношенні подоби.

Структурно-функціональні моделі представляються у вигляді схем, графіків, діаграм, таблиць, рисунків зі спеціальними правилами їх об'єднання і перетворення.

Математичне моделювання здійснюється засобами математики і логіки.

Для концептуального моделювання об'єктами є поняття. Часто під ним розуміють подання системи за допомогою спеціальних знаків, символів,

операцій над ними або за допомогою природних або штучних мов, Але це має відношення лише до форми моделей.

При **імітаційному (програмному)** моделюванні логіко-математична модель досліджуваної системи являє собою алгоритм функціонування системи, реалізований програмою на комп'ютері. Його **предметною (проблемною) областю** якого є виробничі, соціальні, інформаційні, транспортні та інші види систем, проблеми розвитку міст, глобальні світові проблеми та ін. Імітаційне моделювання – потужний засіб рішення задач аналізу і синтезу складних технічних й економічних систем, пошуку ефективних рішень в складних розробках, проектах, програмах розвитку. Воно корисно для рішення **задач** навчання і пізнання, виміру й оцінювання, прогнозування і планування, управління і синтезу.

Об'єктами моделювання в процесі дослідження виступає не уся система, а її елементи, їх відносини, структура, організація, функції, окремі процеси, поведінка суб'єктів, розвиток і т.д. Об'єкт повинен бути коректно описаний і сформульовані цілі моделювання в даній предметній області.

Сформована логіко-математична модель досліджуваної системи являє собою алгоритм функціонування системи, реалізований програмою на комп'ютері. Ця комп'ютерна модель є умовним образом системи об'єктів і процесів, структура яких описується за допомогою таблиць, блок-схем, діаграм, графіків, малюнків, анімаційних фрагментів, гіпертекстів і т.д. Така модель відноситься до виду структурно-функціональних моделей, які були розглянуті у **томі 1**. Але при імітаційному моделюванні ця модель може бути також програмою, яка дозволяє відтворювати (імітувати) процеси функціонування системи і впливу на неї різних, як правило, випадкових, факторів, за допомогою послідовності обчислень і графічного відображення їх результатів, якими є кількісні й якісні характеристики бізнес-системи – її структур, динаміки її розвитку, стабільності, цілісності.

Для розуміння сутності **імітаційної моделі** необхідно знати, чим вона відрізняється від математичної моделі, яка відображає структуру системи і процесів її функціонування у вигляді залежності характеристик системи від її параметрів і параметрів зовнішнього середовища. Імітація процесів функціонування

систем тут зводиться до розрахунків по зазначеним вираженням. А при імітаційному моделюванні процеси функціонування системи імітуються на побудованій моделі, яка їх описує. Таким чином, воно засновано не лише на розрахунковому, як при математичному моделюванні, а і на експериментальному підході.

Методологією імітаційного моделювання є системний аналіз, за допомогою якого будується узагальнені моделі, що відображають всі фактори, властивості і взаємозв'язки реальної бізнес-системи, критерії й обмеження. На відміну від цього, математичне моделювання найчастіше базується на дослідженні операцій, теорії математичних моделей, теорії прийняття рішень, теорії ігор та ін.

Метод імітаційного моделювання дозволяє вирішувати завдання виняткової складності, забезпечує імітацію складних і різноманітних процесів, з великою кількістю елементів. Окремі функціональні залежності у таких моделях можуть описуватися досить громіздкими математичними співвідношеннями. Тому імітаційне моделювання ефективно використовується в завданнях дослідження систем з складною структурою для вирішення конкретних проблем.

Зазначені види моделювання можуть застосовуватися самостійно або одночасно, у деякій комбінації. Має місце взаємопроникнення всіх видів моделювання й симбіоз різних інформаційних технологій. Так, наприклад, на ранніх етапах формування імітаційної моделі використовується концептуальне моделювання. Для цілей опису окремих підсистем моделі, а також у процедурах обробки і аналізу результатів обчислювального експерименту і прийняття рішень використовується логіко-математичне моделювання, включаючи методи штучного інтелекту. З фізичного в імітаційне моделювання привнесена технологія проведення і планування обчислювального експерименту з відповідними математичними методами. Структурно-функціональне моделювання використовується при створенні описів багатомодельних комплексів і при формуванні діаграм імітаційних моделей.

Нові можливості реалізації моделей різного виду (семантичних, логічних, математичних) відкрив розвиток інформаційних технологій.

Змістовним прикладом об'єкту імітаційного моделювання може бути центр обслуговування, подібний тому, що функціонує у банку, довідковій службі або на підприємстві з ремонту автомобілів. Метою керівника цього центру повинно бути зниження витрат шляхом зменшення до мінімуму кількості обслуговуючого персоналу – касирів банку, операторів телефону або автомобільних механіків. Але при цьому потрібно мінімізувати і час, витрачений клієнтами на очікування. Щоб домогтися цього, йому необхідно одержати від моделі наступну інформацію: кількість обслуговуючого персоналу, число клієнтів у черзі на обслуговування і час очікування. Для моделювання процесу обслуговування клієнтів потрібно, щоб фахівець, який створює модель, побудував діаграму системи з потоком вхідних даних, використовуючи середовище розробки, що входить у наявні засоби моделювання.

Роботу даного центру обслуговування можна описати **динамічною дискретною моделлю**, в якій через певні інтервали часу в процес надходять клієнти, стають у чергу, одержують обслуговування і віддаляються. Подією, яка управляє моделлю, служить поява клієнтів. Вхідними даними є число клієнтів, чії потреби слід задовольнити, частота їх появи в центрі, кількість обслуговуючого персоналу і час, витрачений службовцем на одного клієнта. Частота появи клієнтів вводиться в один блок моделі, а кількість обслуговуючого персоналу і час обслуговування клієнта – в іншій. В ході виконання моделі клієнти переходять з блоку в блок з швидкістю, що відповідає часу обслуговування.

Завершивши побудову моделі, розроблювач може приступити до визначення залежності між числом клієнтів, які звернулися за обслуговуванням, і часом очікування в черзі, а також з'ясувати, скільки клієнтів виявилися не обслуженими. Зіставивши отриману інформацію з вимогами, пропонованими до даного центра обслуговування, і припустимими обмеженнями, можна далі в інтерактивному режимі ввести в модель зміни, наприклад, збільшити кількість обслуговуючого персоналу, і знову запустити модель. Ці дії можна повторювати до одержання задовільного результату.

12.1.2 Вимоги до імітаційних моделей

Для побудови і дослідження моделей складних систем необхідно визначити вимоги до них, форми подання і вид опису моделі, характер реалізації моделі і метод дослідження.

Залежно від цільової спрямованості моделі, задаються спеціальні вимоги до самих моделей. Найбільш характерними вимогами до моделей є: цілісність, відбиття інформаційних властивостей, багаторівневність, множинність (багатомодельність), розширюваність, універсальність (абстрактність), можливість побудови самої моделі й її досліджень, можливість матеріалізації моделі у вигляді реальної системи в завданнях проектування.

Модель повинна замінити собою дійсність з тим ступенем абстракції, що корисніше для поставленої мети. Насамперед вона повинна відображати ті істотні властивості і сторони об'єкта, які визначені практичним завданням. Таким чином, модель будується **цілеспрямовано**. Необхідно правильно позначити і сформулювати проблему, чітко задати **мету** дослідження, залежно від якої аналітики і розроблювачі формують моделі реально існуючих об'єктів, або ідеальні моделі ще не існуючих систем, які тільки проектуються. Моделі повинні бути придатними для рішення поставлених реальних завдань. Головною **вимогою до моделей** є їх адекватність реальної дійсності, щоб бути впевненим, що результати точно відображають дійсне положення речей.

Модель повинна бути надійною, простою і зрозумілою користувачеві, а також і технологічною, тобто легкою і зручною в управлінні для особи, яка приймає рішення. Крім того, необхідно, щоб вона була функціонально повною з погляду можливостей рішення необхідних завдань, і адаптивною до змін, дозволяючи легко переходити до інших модифікацій, обновляти дані й удосконалити її в результаті взаємодії з користувачем.

Крім того, модель повинна бути ефективною. Це означає, що витрати часових, трудових, матеріальних ресурсів на побудову моделей і проведення експериментів будуть у припустимих межах або виправдані щодо особливих обставин. В задачах опису, управління, планування важливою вимогою є

адекватність моделі реальним умовам. В задачах проектування і створення унікальних систем найбільш важливим має бути можливість створення на базі моделі реальної системи, наприклад, системи автоматизації проектування (САПР) або системи підтримки прийняття рішень (СППР).

Формами подання моделі є знакові подання (описи, структурні схеми, логічні і математичні конструкції), образи, лабораторні і діючі макети, досвідчені зразки. **Видами опису моделі** знакових логічних форм можуть бути обчислення предикатів, обчислення висловлювань, семантичні сіті, фрейми та інші представлення предметної області. Для математичних форм видами опису є алгебраїчні, диференціальні, інтегральні та інші рівняння. **Характер реалізації** знакових моделей може бути аналітичним, комп'ютерним і фізичним (автоматним).

Вибір методів дослідження залежить від складності моделі, мети моделювання, ступеня невизначеності характеристик моделі. Виділяють розрахунковий, статистичний й імітаційний методи, а також методи самоорганізації досліджень. **Розрахункове моделювання** застосовується при дослідженні математичних моделей і зводиться до їх машинної реалізації при різних числових вихідних даних. Результати цих розрахунків видаються в графічній або табличній формах. Наприклад, машинна реалізація системи диференціальних рівнянь заснована на застосуванні чисельних методів, за допомогою яких математична модель приводиться до алгоритмічного виду і програмно здійснює на комп'ютері розрахунки.

Статистичне моделювання методом Монте-Карло здійснюється відтворенням функціонування імовірнісних моделей, або дослідженням детермінованих процесів, заданих у вигляді математичних моделей з логічними елементами. Його особливістю є випадкове завдання початкових даних з відомими законами розподілу і, як слідство, імовірнісне оцінювання характеристик досліджуваних процесів. Воно застосовується при дослідженні слабо організованих систем з нескладною логікою функціонування.

При **самоорганізуючому моделюванні** функція побудови моделей й її перетворення в процесі експериментування покладається на програмну систему.

12.2 Технологічні особливості імітаційного моделювання

Окремі елементи, процеси можуть описуватися в імітаційній моделі інтегральними, диференціальними та іншими рівняннями і реалізовуватися за допомогою традиційних обчислювальних процедур. Це пов'язано з тим, що інструментарій імітаційного моделювання включає всі можливі засоби, зокрема, арсенал аналітичного моделювання на етапі ідентифікації імітаційної моделі. Аналітичні методи використовуються у стратегічному плануванні обчислювального експерименту і при обробці його результатів. В імітаційному моделюванні постійно зростає роль аналітичних методів.

Імітаційне моделювання містить в собі ідеї і прийоми статистичного моделювання на комп'ютері. Воно підходить для дослідження стохастичних систем, випадкових процесів. На вході використовуються перемінні, які задаються відомими законами розподілу. Можна реалізувати імовірнісний розвиток ситуацій, описувати випадкові процеси, проводити імовірнісне оцінювання характеристик моделі на виході. Тобто в імітаційному моделюванні втілюються ідеї методу Монте-Карло. Воно історично виросло з методу статистичних випробувань. Але в імітаційному моделюванні мова йде про дослідження складних систем і рішення складних проблем, і відображається структура і динаміка моделюємої системи. На комп'ютері реалізуються не статистичні випробування, а проводяться цілеспрямовані обчислювальні експерименти.

Імітаційне моделювання застосовується для дослідження складних логіко-математичних моделей з неточним завданням вихідних даних. Відомі лише закон розподілу, а оцінюючі характеристики визначаються в результаті проведення експериментів на моделі. Тому тут важливу роль грає не тільки проведення, але і планування експериментів.

Особливістю імітаційного моделювання є також те, що імітаційна модель дозволяє відтворювати процеси, які моделюються, з збереженням їх логічної структури і поведінкових властивостей – послідовності чергування в часі подій, що відбуваються в системі, тобто динаміки взаємодій.

При імітаційному моделюванні структура моделюємої системи адекватно відображається в моделі, а процеси її функціонування програються (імітуються) на побудованій моделі. Виділяють статичний опис структури системи, для чого потрібно виконувати структурний аналіз процесів, й опис динаміки взаємодій елементів системи, для створення якого потрібна побудова функціональної моделі моделюємих динамічних процесів.

З погляду програмної реалізації ідея методу імітаційного моделювання полягає в тому, що елементам системи ставляться у відповідність деякі програмні компоненти, а стан цих елементів описується за допомогою перемінних. Моделюючий алгоритм моделює функціонування окремих елементів, які взаємодіють або обмінюються інформацією. Є алгоритм змінення перемінних, що описують стани системи. Динаміка реалізується за допомогою **механізму просування модельного часу**.

Метод імітаційного моделювання дає можливість опису і відтворення взаємодії між різними елементами системи. Щоб створити імітаційну модель треба представити реальну систему (процес), як сукупність взаємодіючих елементів, і алгоритмічно описати функціонування окремих елементів. Після цього треба описати процес взаємодії різних елементів між собою і з зовнішнім середовищем.

Ключовим моментом в імітаційному моделюванні є виділення і опис **станів** системи **набором перемінних станів**, кожна комбінація яких описує конкретний стан. Шляхом зміни значень цих перемінних можна імітувати перехід системи з одного стану в інший. Таким чином, імітаційне моделювання – це подання **динамічної поведінки** системи за допомогою просування її від одного стану до іншого відповідно до певних операційних правил. Ці зміни станів можуть відбуватися або безупинно, або в дискретні моменти часу.

Імітаційне моделювання є динамічне відбиття змін стану системи з часом. Для опису динаміки моделюємих процесів керуюча програма має **механізм завдання модельного часу**. Щоб забезпечити на комп'ютері імітацію паралельних подій реальної системи, вводять деяку глобальну перемінну t , названу **модельним (або системним) часом**. Вона забезпечує синхронізацію всіх подій у системі

Існують два основних способи зміни t : покроковий – з фіксованими інтервалами зміни модельного часу, і подійний, при якому величина кроку вимірюється змінним інтервалом до наступної події.

Покрокове просування часу відбувається з мінімально можливою постійною довжиною кроку. Цей спосіб застосовується в наступних випадках:

- якщо закон зміни перемінних від часу описується інтегро-диференціальними рівняннями, які вирішуються чисельними методами. При цьому динаміка моделі є дискретним наближенням реальних безперервних процесів;

- коли події розподілені рівномірно і можна підібрати крок зміни тимчасової координати;

- коли складно пророчити появу певних подій і коли їх дуже багато і вони з'являються групами.

Цей спосіб не дуже ефективний з погляду використання машинного часу на їх реалізацію. Подійний метод краще застосовувати, коли події розподілені нерівномірно на часовій осі і з'являються через значні часові інтервали. В цьому методі координати часу міняються тільки коли змінюється стан системи. При цьому довжина кроку часового зрушення може бути максимально можливою. Модельний час змінюється від текучого до найближчого моменту настання наступної події. Цей метод кращий, якщо частота настання подій невелика. Тоді більша довжина кроку дозволить прискорити хід модельного часу. На практиці цей метод одержав найбільше поширення. За допомогою розглянутого механізму паралельні процеси, які відбуваються в моделі, перетворюються в послідовні. Такий спосіб подання зветься квазіпаралельним.

Виходячи з розглянутих способів просування модельного часу розрізняють безперервні, дискретні і безперервно-дискретні види імітаційних моделей. В **безперервних** імітаційних моделях стан моделюємої системи міняється як безперервна функція часу і, як правило, ця зміна описується системами диференціальних рівнянь. Відповідно просування модельного часу залежить від чисельних методів рішення диференціальних рівнянь.

В **дискретних** імітаційних моделях перемінні змінюються в певні моменти імітаційного часу (настання подій). Динаміка дискретних моделей являє собою процес переходу від моменту настання чергової події до моменту

настання наступної події. В **безперервно-дискретних** моделях об'єднуються механізми просування часу, які характерні для обох видів моделей.

Логіко-математичні моделі, які використовуються в імітаційному моделюванні складної системи можуть бути як алгоритмічними, так і неалгоритмічними. Для того, щоб вони були машинно-реалізованими, будується **моделюючий алгоритм**, який описує структуру і логіку взаємодії елементів у системі. Програмна реалізація моделюючого алгоритму і є імітаційною моделлю. Вона складається із застосуванням засобів автоматизації моделювання.

Ціль дослідження бізнес-системи полягає в тому, щоб зібрати і проаналізувати в результаті проведення експерименту на імітаційній моделі інформацію про функціонування системи, необхідну для ухвалення рішення. Експериментальна природа імітації визначила походження назви методу. Імітаційне моделювання – є експериментальний метод дослідження реальної системи по її імітаційній моделі. Імітаційні моделі – це моделі прогонного типу, у яких є вхід і вихід. Якщо подати на її вхід певні значення параметрів, то можна одержати результат, дійсний тільки при цих значеннях, закладених в імітаційну програму. Для нових значень параметрів або взаємозв'язків імітаційна програма повинна бути запущена знову, тобто необхідно здійснювати прогін імітаційних моделей, а не вирішувати їх.

Імітаційна модель не здатна формувати своє власне рішення в тому вигляді, як це має місце в аналітичних моделях, а може служити як засіб для аналізу поведінки системи в умовах, які визначаються системним аналітиком. Ця модель – зручний апарат для дослідження **стохастичних систем**. Їх динаміка залежить від випадкових факторів, а вхідні і вихідні перемінні описуються як випадкові величини, функції, процеси, послідовності.

Особливістю моделювання таких систем з урахуванням дії випадкових факторів є те, що шукані величини при дослідженні процесів методом імітаційного моделювання визначають як середні значення щодо великої кількості даних реалізації процесу. Тому експеримент на моделі містить кілька реалізацій, прогонів і припускає оцінювання за даними сукупності (вибірки). При цьому, за законом більших чисел, чим більше число реалізацій, тим одержувані оцінки все більше

здобувають статистичну стабільність. Одного прогону по певним операційним правилам і конкретному набору параметрів досить тільки для **детермінованого** моделювання.

Якщо цілями моделювання є дослідження системи при різних умовах, оцінка альтернатив, знаходження залежності виходу моделі від ряду параметрів і, нарешті, пошук деякого оптимального варіанту, то аналітик, змінюючи значення параметрів на вході моделі, повинен виконати численні машинні прогони імітаційної моделі. Це необхідно для збору, нагромадження і наступної обробки даних про функціонування системи. При цьому виникають проблеми: як зібрати ці дані, як провести серію прогонів і як організувати цілеспрямоване експериментальне дослідження. Внаслідок того, що вихідних даних, отриманих в результаті такого експериментування, може виявитися дуже багато, то виникає проблема їх обробки, яка набагато більш складна, ніж завдання статистичного оцінювання.

Таким чином, перед аналітиком, що використовує методи імітаційного моделювання, завжди встає проблема **організації і планування експерименту**, тобто вибору методу збору інформації для досягнення поставленої мети дослідження, і визначення її обсягу, прагнучи при цьому зменшити витрати часу на експлуатацію моделі за рахунок мінімізації кількості імітаційних прогонів. Для її рішення використовуються методи регресійного аналізу і планування експерименту.

При **стратегічному** плануванні з'ясовується взаємозв'язок між керованими перемінними або знаходити комбінацію значень керованих перемінних, яка дає екстремальний відгук (вихід) імітаційної моделі. **Тактичне** планування пов'язане з визначенням способів проведення намічених імітаційних прогонів. Тут вирішуються завдання визначення тривалості прогону, оцінка точності результатів моделювання та ін. Експерименти з імітаційною моделлю, зміст яких визначається попередньо проведеним аналітичним дослідженням, а результати достовірні і математично обґрунтовані, називаються **спрямованими обчислювальними експериментами**.

12.3 Основні етапи імітаційного моделювання

12.3.1 Склад етапів

Виділяються наступні етапи імітаційного моделювання:

1.Формулювання проблеми і визначення цілей імітаційного дослідження.

2.Розробка концептуального опису. Результатом є концептуальна модель і вибір способу формалізації для заданого об'єкта моделювання.

3.Формалізація імітаційної моделі. Складається формальний опис об'єкта моделювання.

4.Програмування імітаційної моделі (розробка програми-імітатора). **Вибираються** засоби автоматизації моделювання, виконується алгоритмізація, програмування і налагодження імітаційної моделі.

5.Випробування і дослідження моделі, перевірка моделі. Проводиться верифікація моделі, оцінка адекватності, дослідження властивостей імітаційної моделі та інших процедур комплексного тестування розробленої моделі.

6.Планування і проведення імітаційного експерименту. Результатом є складений план експерименту, задані умови імітаційного прогону для обраного плану і їх реалізація.

7.Аналіз результатів моделювання. Проводиться інтерпретація результатів моделювання і їх використання для прийняття рішень.

12.3.2 Формулювання проблеми і цілей дослідження

На цьому етапі визначається і детально вивчається об'єкт моделювання, особливо ті сторони його функціонування, які становлять інтерес для дослідження. Потім формулюється проблема і визначаються цілі дослідження. Від формулювання дослідником проблеми і визначення цілей, які повинні бути досягнуті в результаті імітації, залежить ухвалення рішення про доцільність застосування методу імітаційного моделювання і, значною мірою, вибір типу імітаційної моделі і характер наступного дослідження.

Результатом робіт на даному етапі є змістовний опис об'єкта моделювання з вказівкою проблеми і цілей імітації, а також і тих аспектів функціонування об'єкта моделювання, які треба вивчити на імітаційній моделі. Змістовний опис складається в термінології реальної системи, мовою предметної області, зрозумілому замовникові. В ході складання змістовного опису об'єкта моделювання встановлюються межі вивчення об'єктів, що моделюється, дається опис зовнішнього середовища, з яким він взаємодіє. Формулюються також основні критерії ефективності, щодо яких передбачається проводити порівняння на моделі різних варіантів рішень, проводиться генерація й опис розглянутих альтернатив.

Загальна послідовність дій на цьому етапі наступна:

- збір даних про об'єкт моделювання і складання **змістовного опису об'єкта моделювання**;

- вивчення проблемної ситуації, визначення діагнозу і постановка задачі;

- уточнення цілей моделювання;

- обґрунтування необхідності моделювання і здійснення вибору методу моделювання;

- формування концептуальної моделі досліджуваного об'єкта.

Сучасний теоретичний апарат опису систем поки не може гарантувати оптимальність результатів моделювання для поставлених цілей. При побудові моделей доводиться покладатися на мистецтво, досвід, інтуїцію. Системний аналітик повинен уміти аналізувати проблему, починаючи з детального вивчення всіх аспектів функціонування. З позицій системного підходу треба розкрутити клубок проблем від початкового їх формулювання з розширенням до проблематики. Загальне завдання моделювання при цьому розбивається на частки. Важливо правильно поставити завдання. При цьому слід врахувати, що вивчення конкретних зразків моделей не сприяє розвитку творчих здатностей до створення моделей.

Системний розгляд сутності проблеми включає в себе:

- обґрунтування сутності і місця досліджуваної проблеми;

- формування структури досліджуваної системи;

- виявлення повного набору значущих факторів;

- визначення функціональних залежностей між факторами.

Етапами побудови концепції вирішення проблеми є:

- дослідження об'єктивних умов рішення проблеми;
- обґрунтування цілей, завдань, необхідних для вирішення проблеми, структуризація завдань, формалізація цілей;
- вибір або розробка засобів рішення проблеми: опис альтернатив, сценаріїв, правил і керуючих впливів для відпрацьовування на моделі процедур прийняття рішень.

В завершення виконується структуризація завдань моделювання, аналізуються можливості методів моделювання і здійснюється вибір ефективних методів моделювання.

Перш за все треба визначити цільове призначення моделі. Далі може бути застосований метод декомпозиції цілей, що припускає поділ цілого на частині: цілей – на підцілі, завдань – на підзавдання і т.д. В результаті будується ієрархічна деревоподібна структура (дерево цілей). Вся ця процедура виконується експертами по проблемі і фахівцями. Тому тут є присутнім суб'єктивний фактор. Кожний експерт зробить по-своєму. Практичні результати будуть залежати від того, наскільки повно все було структуроване. Те, що для одного рівня ціль, для іншого – засіб. Часто їх плутають і відбувається змішання цілей. Ціль – є опис бажаного майбутнього, тому тут легко помилитися. Для складної системи цілі можуть бути суперечливими. Ціль рідко буває єдиною. Можливо невірне ранжирування цілей.

Цілями моделювання можуть бути [49]:

- оцінка – визначення, наскільки гарно запропонована структура система буде відповідати конкретним критеріям;
- порівняння альтернатив – зіставлення конкуруючих систем для виконання певної функції, або зіставлення декількох пропонуванних робочих методик;
- прогноз – оцінка поведінки системи при деякому передбачуваному сполученні робочих умов;
- аналіз чутливості – виявлення тих факторів, які найбільше впливають на поведінку системи;
- виявлення функціональних співвідношень – визначення природи залежності між факторами і відгуком системи.
- оптимізація – точне визначення такого сполучення діючих факторів і їх величин, при якому забезпечується найкращий відгук всієї системи в цілому до заданого критерію.

Дуже важливо чітко і однозначне визначення критеріїв. Розрізняють критерії, за допомогою яких оцінюється ступінь досягнення цілі системою, і критерії, за якими оцінюється спосіб руху до цілі або ефективність засобу досягнення цілей. Для багатокритеріальних завдань набір критеріїв необхідно структурувати за підсистемами і ранжировати за їх важливістю.

12.3.3 Розробка концептуальної моделі об'єкта моделювання

Концептуальна модель являє собою логіко-математичний опис моделюємої системи, відповідно до сформульованої проблеми. На цьому етапі треба сформулювати загальний задум моделі і перехід від реальної системи до логічної схеми її функціонування. Виконується опис об'єкта в термінах математичних понять і алгоритмізації функціонування його компонент. Концептуальний опис – це спрощене алгоритмічне відображення реальної системи. При його розробці визначається основна структура моделі, яка включає **статичний і динамічний** опис системи. Крім того, визначаються межі системи, приводиться опис зовнішнього середовища і визначаються її впливи на систему. Далі виділяються істотні елементи і дається їх опис, формуються перемінні, параметри, функціональні залежності як для окремих елементів і процесів, так і для всієї системи, обмеження, цільові функції (критерії). Результат роботи на цьому етапі – документований концептуальний опис і обраний спосіб формалізації моделюємої системи. На цьому етапі уточнюється також методика всього імітаційного експерименту. При створенні невеликих моделей цей етап сполучається з етапом складання змістовного опису системи.

Побудова концептуальної моделі починається з висування гіпотез і фіксації всіх необхідних допущень. Треба пам'ятати, що визначення системи завжди суб'єктивно. Воно залежить від мети моделювання, і від того, хто саме визначає систему. Далі здійснюється **декомпозиція** системи. Визначаються найбільш істотні елементи системи і взаємодії між ними з погляду сформульованої проблеми, тобто виконується **структурний аналіз** системи. Виявляються

основні аспекти її функціонування і складається **функціональна модель**, приводиться **опис зовнішнього середовища**.

Складна система розбивається на частині, зберігаючи при цьому зв'язки, які забезпечують взаємодію. Важливо визначити, які компоненти будуть включені в модель, які будуть винесені в зовнішнє середовище, і які взаємозв'язки будуть установлені між ними.

Опис зовнішнього середовища повинний відбити вплив його елементів на елементи системи. Обговорюється рівень деталізації процесів, які моделюються. При цьому важливо розуміти, що в підставі всякої декомпозиції лежать два суперечливих принципи: **повнота і простота**. На початкових етапах складання моделі спостерігається тенденція до урахування надмірно великої кількості компонентів і перемінних. Однак відомо, що ступінь розуміння явища обернено пропорційна числу перемінних, що фігурують у його описі. Модель, перевантажена деталями, може стати надмірно складною і її важко буде реалізувати.

Треба прагнути робити спочатку прості моделі, а потім їх ускладнювати. Компроміс тут полягає в тому, що в модель включаються тільки компоненти, істотні стосовно цілі аналізу. Необхідно додержуватися принципу **ітеративної** побудови моделі, коли в міру вивчення системи по моделі, в ході розробки, модель змінюється шляхом додавання нових або виключення деяких її елементів і/або взаємозв'язків між ними.

Основними прийомами моделювання є спрощення й абстракція. Обраний рівень деталізації повинен дозволяти абстрагуватися від неточно сформульованих певних аспектів функціонування реальної системи внаслідок недоліку інформації. Під **спрощенням** розуміється зневага несуттєвими деталями або прийняття припущень про більш прості співвідношення, наприклад, припущення про лінійну залежність між перемінними.

При моделюванні висуваються гіпотези, припущення, що мають відношення до взаємозв'язку між компонентами і перемінними системи. **Абстракція** містить в собі істотні якості поведінки об'єкта, але не обов'язково в тій же формі і настільки детально, як у реальній системі.

Після аналізу і моделювання частин або елементів системи, треба перейти до їх **синтезу**, тобто до об'єднання в єдине ціле. В концептуальній моделі повинна бути коректно відбита взаємодія елементів. При цьому уточнюється структура, упорядкування і опис відносин.

Системне дослідження побудоване на сполученні операцій аналізу і синтезу. На практиці реалізуються ітеративні процедури аналізу і синтезу. Лише після цього можна намагатися пояснити ціле – систему, через її підсистеми у вигляді загальної структури цілого.

В опис системи повинні бути включені **критерії ефективності** функціонування системи й оцінюваних альтернативних рішень. Останні можуть розглядатися як входи моделі або сценарні **параметри**. При алгоритмізації процесів, які моделюються, уточнюються також основні їх **перемінні**. Кожна модель являє собою деяку комбінацію компонентів, перемінних, параметрів, функціональних залежностей, обмежень, цільових функцій (критеріїв) [49].

Під **компонентами** розуміють складові частини, які при відповідному об'єднанні створюють систему. Іноді компонентами вважають також **елементи** системи або її **підсистеми**. Система визначається як група або сукупність об'єктів, об'єднаних деякою формою регулярної взаємодії або взаємозалежності для виконання заданої функції.

Параметрами є величини, які дослідник може вибирати довільно, на відміну від **перемінних** моделі, які можуть приймати тільки значення, обумовлені видом даної функції. У моделі розрізняють перемінні двох видів: екзогенні й ендогенні. **Екзогенні** перемінні породжуються поза системою або є результатом взаємодії зовнішніх причин. Ендогенні перемінні виникають в системі в результаті впливу внутрішніх причин. В тих випадках, коли ендогенні перемінні характеризують стан або умови в системі, вони називаються **перемінними стану**. Коли ж необхідно описати входи і виходи системи, то їх називають **вхідними і вихідними перемінними**.

Функціональні залежності описують поведінку перемінних і параметрів в межах компонента або ж виражають співвідношення між компонентами системи. Ці співвідношення по своїй природі є або детерміністськими, або стохастичними.

Обмеження являють собою встановлювані межі зміни значень перемінних або обмежуючі умови їх змін. Вони можуть вводитися або розроблювачем, або встановлюватися, виходячи із властивостей самої системи.

Цільова функція (функція критерію) точно відображає цілі або завдання системи і необхідні правила оцінки їх виконання. Вираження для цільової функції повинно бути однозначним щодо визначених цілей і завдань, з якими повинні порівнюватися прийняті рішення.

12.3.4 Формалізація імітаційної моделі

Процес формалізації складної системи включає вибір способу формалізації і складання формального опису системи. В процесі побудови моделі виділяють три рівні її подання: неформалізований (концептуальна модель), формалізований (формальна модель) і програмний (імітаційна модель). Кожний рівень відрізняється від попередніх ступенем деталізації системи і способами опису її структури і процесу функціонування. При цьому рівень абстрагування зростає.

Концептуальна модель є систематизованим змістовним описом системи або проблемної ситуації неформальною мовою. Він включає визначення основних елементів системи, їх характеристики і взаємодії між елементами власною мовою. Можуть використовуватися таблиці, графіки, діаграми і т.д. Цей опис моделі необхідний як самим розроблювачам (при перевірці адекватності моделі, її модифікації і т.д.), так і для взаєморозуміння з фахівцями інших профілів. Концептуальна модель містить інформацію для системного аналітика, який виконує формалізацію системи на базі певної методології і технології.

На підставі неформалізованого опису здійснюється розробка формалізованого опису, який далі буде перетворюватися в програму-імітатор відповідно до технології програмування. Аналогічна схема має місце і при виконанні імітаційних експериментів: змістовне формулювання відображається

на формальну модель, після чого вносяться необхідні зміни і доповнення в методику спрямованого обчислювального експерименту. Формальна модель складної системи повинна бути вільна від другорядної інформації, наявної в змістовному описі, і **алгоритмічно** представляти об'єкт моделювання. Наприклад, система диференціальних рівнянь перетвориться в алгоритмічну з використанням чисельного методу інтегрування, при цьому властивості моделі міняються і це треба враховувати.

Треба відбити на рівні моделюючого алгоритму взаємодії компонентів між собою. Може виявитися, що інформації, наявної в змістовному описі, недостатньо для формалізації об'єкта моделювання. В цьому випадку необхідно повернутися до етапу складання змістовного опису і доповнити його необхідними даними. На практиці таких повернень може бути декілька. Формалізація корисна в певних межах і для простих моделей невиправдана. Є різноманітні концепції формалізації і структуризації, які знайшли застосування в імітаційному моделюванні. Вони орієнтуються на різні математичні теорії і виходять із різних подань про досліджувані процеси.

Для дискретних моделей можуть застосовуватися процесно-орієнтовані системи (process description), системи, засновані на мережних парадигмах (network paradigms), потокові діаграми моделей системної динаміки. Найбільш відомі і широко використовуються на практиці такі концепції формалізації: агрегатні системи і автомати, мережі Петрі, застосовувані при структуризації причинних зв'язків і моделюванні систем з паралельними процесами, що необхідно для стратифікації й алгоритмізації динаміки дискретних і дискретно-безперервних систем і їх розширення, моделі системної динаміки. Для кожної концепції формалізації можуть бути реалізовані різноманітні **алгоритмічні моделі**.

Як правило, та або інша концепція структуризації або формалізації на технологічному рівні закріплена в системі моделювання і мові моделювання, і підтримується спеціально розробленими прийомами технології програмування. Це спрощує побудову і програмування моделі. Наприклад, мова моделювання GPSS має блокову концепцію структуризації. Структура процесу, що

моделюється, зображується потоком транзактів, що проходить через обслуговуючі устрої, черги та інші елементи систем масового обслуговування. В ряді сучасних систем моделювання є і спеціальні засоби, що забезпечують застосування в системі певної концепції формалізації.

Транзакт – це формальний запит на яке-небудь обслуговування, це динамічна одиниця будь-якої моделі. Транзакт відрізняється від звичайних заявок, котрі розглядаються при аналізі моделей масового обслуговування, мають набір особливих властивостей і параметрів, які динамічно змінюються. Шляхи міграції транзактів по графу стохастичної сіті визначаються логікою функціонування компонентів моделі у вузлах сіті.

Транзакт може виконувати наступні функції:

- породжувати групи (сімейства) інших транзактів;
- вбирати інші транзакти конкретного сімейства;
- захоплювати ресурси і використовувати їх деякий час, а потім звільняти;
- визначати час обслуговування, накопичувати інформацію щодо пройденого шляху і мати інформацію щодо свого наступного шляху і шляхах інших транзактів;
- мігрувати в модельному просторі.

Приклади транзактів: вимога на переказ грошей, замовлення на виконання робіт в фірмі, наказ керівника, покупець в магазині, пасажир транспортного засобу.

Концептуальний або формальний опис моделі складної системи необхідно перетворити в **програму-імітатор**, відповідно до деякої **технології програмування**, із застосуванням мов і систем моделювання. Для реалізації імітаційної моделі треба вибрати відповідні інструментальні засоби.

12.3.5 Збір і аналіз вхідних даних

Якщо програмування і трасування імітаційної моделі можна виконувати на гіпотетичних даних, то майбутнє експериментальне дослідження необхідно здійснити на реальних потоках даних. Від цього залежить точність одержуваних результатів моделювання. Одержати інформацію про реальну систему можливо з існуючої документації по системі, наприклад, з технічної і фінансової документації

виробничих систем. Часто для завдання вхідної інформації необхідно провести натурні експерименти на системі або її прототипах. Такий підхід застосовується для космічних, військових досліджень, в авіації. У більш простих випадках можна проводити, наприклад, хронометраж при виконанні виробничих операцій.

Іноді потрібні дані можуть не існувати, і сама природа системи виключає можливість фізичного експериментування, наприклад, проектні системи, прогнозування в соціально-економічних дослідженнях. В цьому випадку використовуються різні процедури, засновані на загальному аналізі проблематики, анкетуванні, інтерв'юванні, експертному оцінюванні. При моделюванні інформаційних систем тривалість виконання інформаційної вимоги оцінюється на підставі даних щодо реалізованих алгоритмів. Для стохастичних систем виникають труднощі щодо невідомості законів розподілу і визначення імовірнісних характеристик (середніх значень, дисперсій, кореляційних функцій і т.п.) для аналізованих процесів і їх параметрів.

Необхідність статистичного аналізу при зборі й аналізі вхідних даних пов'язана з потребою визначення виду функціональних залежностей, що описують вхідні дані, оцінкою конкретних значень параметрів цих залежностей, а також перевіркою значимості параметрів. Для підбора теоретичних розподілів випадкових величин застосовують відомі методи математичної статистики, засновані на визначенні параметрів емпіричних розподілів і перевірці статистичних гіпотез. При цьому використовуються критерії узгодження наявних емпіричних даних з відомими законами розподілу на статистично прийнятному рівні, що є довірливим.

12.3.6 Випробування і дослідження властивостей імітаційної моделі

Після того, як імітаційна модель реалізована на комп'ютері, необхідно перевірити вірогідність моделі. Для цього проводиться комплексне **тестування** моделі. Це – планований ітеративний процес, спрямований на підтримку процедур верифікації і валідації імітаційних моделей. Деякі корисні процедури тестування можна знайти в роботах [15, 27, 52].

Якщо в результаті проведених процедур **імітаційна** модель виявиться недостатньо достовірною, то її треба **калібрувати** . Для цього в моделюючий алгоритм вбудовуються калібровані коефіцієнти. Для складних моделей можливі численні ітерації на ранні етапи щоб одержати додаткову інформацію про об'єкт, який моделюється, або для їх доробки.

Наявність виявлених помилок у взаємодії компонентів моделі повертає аналітика до етапу створення імітаційної моделі. Можливо, в ході формалізації він занадто спростив фізичні явища, виключив з розгляду ряд важливих сторін функціонування системи, що привело до неадекватності моделі об'єкта. В тих випадках, коли виявився невдалим вибір способу формалізації, аналітикові необхідно повторити етап складання концептуальної моделі з урахуванням нової інформації і досвіду, що з'явився. Нарешті, коли у аналітика виявилось недостатньо інформації про об'єкт, він повинен повернутися до етапу складання змістовного опису системи і уточнити його.

Організація **спрямованого обчислювального експерименту** на імітаційній моделі припускає вибір і застосування різних аналітичних методів для обробки результатів імітаційного дослідження: методи планування обчислювального експерименту, регресійний і дисперсійний аналіз, градієнтні та інші методи оптимізації. Проведене дослідження повинне дозволити зробити висновки за отриманими результатами, достатні для прийняття рішень по позначеним на ранніх етапах проблемам і завданням.

З розглянутого видно, що процес імітаційного дослідження є трудомістким ітеративним процесом. Він вимагає, щоб розроблювач моделей був і мистецьким системним аналітиком, і майстром-технологом, що володіє сучасними комп'ютерними технологіями, застосовуваними при створенні і дослідженні імітаційних моделей. Він повинен мати хорошу аналітичну підготовку, яка дозволяє коректно застосовувати методи математичної статистики та інші математичні і обчислювальні процедури як для цілей ідентифікації імітаційних моделей, так і для обробки результатів експериментального дослідження.

12.4 Моделі і методи структуризації і формалізації імітаційних систем

2.4.1 Підходи до побудови імітаційних моделей

В практиці аналізу складних систем широко використовуються наступні моделі і методи структуризації і формалізації імітаційних систем:

- транзактно-орієнтований підхід мови моделювання дискретного типу GPSS;
- мережі кусково-лінійних агрегатів, які моделюють дискретні і безперервно-дискретні системи;
- мережі Петрі і їх розширення, які застосовуються при структуризації причинних зв'язків і моделюванні систем з паралельними процесами, і можуть служити для стратифікації і алгоритмізації динаміки дискретних і дискретно-безперервних систем;
- поточкові діаграми і кінцево-різницеві рівняння системної динаміки, що є моделями безперервних систем.

В дискретних імітаційних системах зміна складу і стану відбувається в дискретні моменти часу, які називаються подіями. Подія – це миттєва зміна стану моделі, що відбулася в результаті здійснення взаємодій між компонентами моделі в той самий момент імітаційного часу.

Функціонування дискретної системи можна описати, визначаючи зміни стану системи, що відбуваються в моменти здійснення подій, фіксуючі дії, в яких беруть участь елементи системи, описуючи процес, через який проходять елементи. Процес – це орієнтована в часі послідовність подій, що може складатися з декількох дій.

Є три альтернативних підходу до побудови дискретних імітаційних моделей: подійний, процесно-орієнтований, який включає транзактний спосіб імітації, і підхід сканування активності. Основа цих підходів заставляється в деякі. Прикладами мов і систем моделювання для цих підходів можуть служити: GASP, SIMSCRIPT, орієнтовані на події, GPSS, SIMULA, призначені для опису паралельних процесів.

12.4.2 Мова моделювання GPSS

В 1961 р. Д. Гордон розробив мову моделювання GPSS (General Purpose Simulating System). Він визначив сучасні технологічні тенденції в дискретному імітаційному моделюванні і з'явився провісником сучасних мов і систем моделювання дискретного типу, таких як Extend, Arena, Process Model, Taylor, WITNESS і сотень інших сучасних комерційних симуляторів. Цьому сприяла, насамперед, вдало сформована базова схема структуризації, закладена в GPSS. Вона підтримує **блоко-орієнтований підхід**, в рамках якого моделюючий блок має своє функціональне призначення і представлений відповідними функціональними об'єктами (аналогічні елементам систем масового обслуговування). Крім того, ця схема має можливості для опису паралельних процесів. Вона дозволяє реалізувати ідеографічний режим формування дискретної моделі з стандартних функціональних блоків. Зв'язки на цих графічних конструкціях інтерпретуються як маршрути проходження рухливих об'єктів в системі. Освоюючи концепції структуризації мови моделювання GPSS, легко зрозуміти ідею і принципи роботи сучасних комерційних симуляторів дискретного типу та інших нових інтелектуальних середовищ.

Сучасний ринок інформаційних технологій представлений такими варіантами технологічного розвитку базової мови GPSS: корпорація Wolverin – GPSS/H з сучасною її мовою SLX, корпорація Minuteman Software – GPSS World, рішення Стокгольмської школи вищої економіки – Micro GPSS, Web GPSS.

В мові GPSS **концепція структуризації** процесу розроблена з орієнтацією на засоби систем масового обслуговування (СМО). Структура процесу зображується у вигляді потоку, що проходить через обслуговуючі пристрої (ОУ), черги, ключі та інші елементи СМО. Модель має блокову структуру. Процес представляється як потік заявок в системі обслуговування. Блоки інтерпретуються як ОУ. Заявки (транзакти) конкурують між собою за місце в ОУ, утворюють черги перед ОУ, якщо вони зайняті. Дуги на блок-схемі – це потенційні потоки заявок між ОУ. Існують джерела і стоки цих заявок. В цьому випадку блок-схема моделі описує маршрути руху заявок в системі. GPSS є гнучким мовним середовищем і дозволяє моделювати не тільки СМО, але й інші системи.

12.4.3 Системи масового обслуговування

Система масового обслуговування – абстрактний об'єкт, в якому виконується послідовність операцій. Він містить у собі сукупність приладів обслуговування, які зв'язані певним логічним порядком. Відповідно до цієї логіки відбувається рух матеріальних носіїв – заявок на обслуговування від каналу (ОУ) до каналу (ОУ). Заявка характеризується моментом появи на вході системи, статусом стосовно інших заявок, деякими параметрами, що визначають потреби в часових ресурсах на обслуговування. Заявки, що постійно поступають на обслуговування, створюють потік заявок, який є сукупністю заявок, що розподілені в часу.

Потік заявок може бути однорідним, коли всі заявки рівноправні з погляду обслуговування, і неоднорідним. Основний параметр потоку заявок – проміжок часу між моментами надходження двох сусідніх заявок. Потік заявок може бути стаціонарним і нестаціонарним, коли він змінюється, залежно від часу доби. Потік заявок розглядається як випадковий процес, який характеризується функцією розподілу періоду надходження заявок, наприклад, найпростіший потік, потік Эрланга.

Елемент системи, в якому відбуваються операції, називається обслуговуючим устроєм. В момент виконання операцій він зайнятий, інакше – вільний. Якщо ОУ (канал) вільний, то заявка приймається до обслуговування. Кожна заявка затримується в ОУ на час, рівний періоду її обслуговування, після якого вона залишає прилад обслуговування. Таким чином, ОУ характеризується часом обслуговування заявки. При випадковому характері надходження утворюються черги. Заявки можуть прийматися до обслуговування по черзі (FIFO, черги з пріоритетами та ін.), у випадковому порядку, відповідно до заданих розподілів, за мінімальним часом одержання відмови.

Реальний процес функціонування СМО представляється у вигляді послідовності фаз обслуговування, виконуваних різними устроями: обслуговування покупців у магазині (прилавок, каса), процес обробки деталей на верстатах. Ці багатофазні системи можуть мати складну структуру (стохастичні мережі). Обслужена заявка залишає прилад обслуговування і

залишає систему (поглинач заявок), або рухається далі відповідно до технологічної схеми роботи системи.

Розрізняють наступні типи СМО:

- одноканальні і багатоканальні (по кількості ОУ);
- с очікуванням і без очікування (з відмовами);
- с обмеженням на довжину черги (або з обмеженим очікуванням) і без обмеження;
- с упорядкованою чергою і з неупорядкованою чергою;
- з пріоритетами і без пріоритетів.

Показниками якості обслуговування є:

- загальна кількість обслужених заявок за проміжок часу;
- пропускна здатність (середнє число заявок, обслужених в одиницю часу);
- частки заявок обслужених і відмовлених;
- час перебування заявки в системі від моменту її надходження в систему до моменту завершення її обслуговування;
- середня довжина черги і середні часи обслуговування й очікування;
- завантаження каналів, яке характеризує ступінь простою ОУ.

Аналітичні методи моделювання СМО мають обмежені можливості для рішення практичних завдань, наприклад, часто використовується припущення про найпростіший потік заявок. Однак для різних фаз обслуговування він може бути не найпростішим.

В імітаційному моделюванні подібні та інші обмеження знімаються. В ньому можуть застосовуватися довільні закони розподілу, різні схеми обслуговування.

Для СМО сутність методу імітаційного моделювання полягає в наступному. Тут СМО не обов'язково досліджується в стаціонарному режимі, а можливе вивчення перехідного режиму, коли показники відрізняються від граничних асимптотичних значень. Використовуються спеціальні алгоритми, що дозволяють виробляти випадкові реалізації потоків подій і моделювати процеси функціонування обслуговуючих систем. Далі здійснюється багаторазове відтворення, реалізація випадкових процесів обслуговування і на виході моделі – статистична обробка отриманих статистичних даних і оцінка показників якості обслуговування.

12.4.4 Система моделювання GPSS

GPSS – це система дискретного типу, яка транзактно-орієнтована на клас об'єктів, процес функціонування яких можна представити у вигляді множини станів і правил переходу з одного стану в інший, обумовлених у дискретній просторово-часовій області. Для реєстрації змін у часі існує таймер модельного часу. Механізм завдання модельного часу - подійний, зі змінним кроком. Зміни в реальній системі приводять до появи подій. Подія – зміна стану будь-якого елемента системи. В системі відбуваються такі події, як надходження заявки, постановка заявки в чергу, початок обслуговування, кінець обслуговування та ін.

В GPSS розглядаються **основні** події, які можна запланувати, тобто розрахувати момент їх настання заздалегідь до їх появи, наприклад, момент появи заявки на вході, і **допоміжні** події, які відбуваються внаслідок появи основних подій і здійснюються в результаті взаємодії таких абстрактних елементів як блоки і транзакти, наприклад, зміна стану приладу обслуговування з “вільний” на “зайнято”.

Прикладами об'єктів, що моделюється, в GPSS є: транспортні об'єкти, склади, виробничі системи, магазини, торговельні об'єкти, комп'ютерні мережі, системи передачі повідомлень. Алгоритмічна схема може бути використана для оформлення складних формальних схем: СМО і стохастичні мережі, автомати, мережі Петрі, автомати та ін.

Функціональна структура GPSS представлена рівнем, який визначається комбінацією основних функціональних об'єктів, таких, як устрої, пам'яті, ключі (логічні перемикачі), черги, транзакції, і рівнем з апаратно-орієнтованими об'єктами, статистичними і обчислювальними об'єктами.

Транзакти є абстрактними рухливими елементами. Вони відображають реальні заявки на обслуговування і є аналогами різних реальних об'єктів: повідомлень, транспортних засобів, людей, деталей і т.д. Транзакти рухаються по моделі, з'являючись в ній з тією ж інтенсивністю, що і реальні заявки. Вони можуть створюватися і знищуватися. Переміщаючись між блоками моделі відповідно до логіки моделювання, транзакти викликають (і випробовують) різні дії. Можливі їх затримки в деяких місцях моделі, пов'язані з обслуговуванням, очікуванням у черзі, зміни маршрутів і напрямку руху, створення копій транзактів і т.д.

При генерації транзактів резервуються 12 параметрів-атрибутів. В їх набір входить його номер, номер блоку, у якому транзакт перебуває в цей момент, номер наступного блоку, час переходу в наступний блок, пріоритет, що характеризує черговість обробки транзактів в певних ситуаціях та ін. При програмуванні можна привласнити транзакту набір параметрів, які виражають властивості або характеристики об'єктів, що моделюється: вага, швидкість, колір, час обробки і т.п.

Устрої моделюють об'єкти, в яких може відбуватися обробка транзакцій, що пов'язане з витратами часу. Устрої є аналогами каналів СМО: кожний устрій в цей момент часу може бути зайнятим лише одним транзактом. Устрій може бути перервано. В GPSS існує можливість перевірки стану устрою.

Пам'яті призначені для моделювання об'єктів, що володіють ємкістю. Пам'ять може обслуговувати одночасно декілька транзактів. При цьому транзакт займає певну частину пам'яті.

Логічні перемикачі приймають значення включене/виключено, дозволяють змінювати шляхи проходження транзактів в моделі. **Статистичні об'єкти** використовуються тільки тоді, коли необхідно збирати статистику. До них ставляться черги і таблиці. Об'єкт **черги** використовують, якщо необхідно збирати інформацію про довжину черги транзактів і часу їх затримки у певних місцях моделі.

Таблиці обробляють статистичну інформацію, будують гістограму розподілів по будь-якій перемінній.

Обчислювальними об'єктами є матриці, функції, перемінні різних типів.

Другий рівень – блок-схема моделі, складена з типових блоків, між якими переміщуються транзакти. **Блоки** – це операційні об'єкти GPSS, в яких реалізуються всі дії, пов'язані з обслуговуванням транзакту (їх створення і знищення, зміна параметрів транзакту, керування потоками транзактів, і т.д.). Кожний блок має стандартне позначення. Послідовність блоків – це послідовність операторів мовою **GPSS**.

Блоки мають вхід-вихід і виконуються тільки в результаті входу в них транзактів. Існує блок **GENERATE**, що має тільки вихід, через який транзакти входять у модель, і блок **TERMINATE**, що має тільки вхід, який видаляє

транзакти з моделі. Опис паралельних процесів являє собою кілька ланцюжків блоків, взаємодіючих через загальні ресурси або перемінні.

Отже, модель системи представляє мережу блоків (операторів мови). Кожний блок описує певний етап дій у системі. Лінії з'єднання блоків показують напрямки руху елементів (транзактів) через систему або описують деяку послідовність подій, що відбуваються в системі.

Є різні узагальнення розглянутої концепції структуризації, коли структура процесу зображується у вигляді потоку, що проходить через обслуговуючі устрої та інші елементи СМО: мережі черг, графи потоків, структурно-стохастичні графи та ін. Дуги на графах інтерпретуються як потенційні потоки заявок між обслуговуваними устроями. Шляхи на графах відповідають маршрутам руху заявок в системі обслуговування. Докладніше вивчення мови моделювання можна виконати за допомогою таких джерел: [6, 33, 41, 50, 59, 63].

12.4.5 Агрегативні моделі

Кусково-лінійний агрегат. В 60-х роках Н.П.Бусленко і И.Н.Коваленко [3, 4, 40] визначили клас **агрегативних** моделей складних систем. В [4] досліджувалися їх структурні і поведінкові властивості. Для цього була створена агрегативна імітаційна система АІС. Основним елементом побудови таких моделей був **кусово-лінійний агрегат (КЛА)**, який відноситься до класу об'єктів, які функціонують у часі $t \in T$. Він здатний сприймати **вхідні сигнали** зі значеннями $x \in X$, видавати **вихідні сигнали** зі значеннями $y \in Y$ і перебувати в кожний момент часу в деякому **стані** $z \in Z$. Ці об'єкти відрізняє специфіка множин X , Y , Z , припустимі форми вхідних і вихідних повідомлень, тобто функцій $x(t)$ і $y(t)$, траєкторій $z(t)$, а також спосіб перетворення вхідного повідомлення у вихідне.

Динаміка КЛА носить “подійний” характер. Події можуть бути внутрішніми і зовнішніми. Внутрішніми подіями є досягнення траєкторією КЛА деякої підмножини станів $Z^* \in Z$, зовнішніми – надходження вхідного сигналу. Стан КЛА змінюється між подіями детермінованим образом. Кожному

стану z ставиться у відповідність величина $x = x(z)$. Вона трактується як потенційний час до настання чергової внутрішньої події.

Стан КЛА в момент t^* – виникнення **внутрішньої** випадкової події. В цей момент видається вихідний сигнал y^* , утримування якого залежить лише від z^* . Вихідний сигнал може бути і порожнім, тобто не видаватися. Після випадкового стрибка $x(z)$ знову визначається час до настання внутрішньої події. В момент t^{**} виникнення зовнішньої події, пов'язаної з надходженням вхідного сигналу, стан КЛА є випадковим, залежним лише від x і z^{**} . В цей момент видається вихідний сигнал y^{**} , зміст якого визначається x і z^{**} . Вхідні сигнали мають **пріоритет** над внутрішніми подіями, тобто якщо моменти настання зовнішніх і внутрішніх подій збігаються, то зміна стану здійснюється відповідно до правила настання зовнішньої події.

Динаміку КЛА можна представити в наступному виді. Нехай у деякий момент був заданий стан КЛА. Тоді визначається час, через який відбувається **випадковий стрибок** $x(z)$, і міняється стан. Починаючи з моменту настання події (зовнішня або внутрішнього), ситуація повторюється, і динаміку КЛА можна описати у вигляді **завдання фазової траєкторії зміни станів** $z(t)$. Процес функціонування КЛА повністю визначається змінами, що відбуваються в моменти настання подій. Між цими моментами стан КЛА міняється **детерміновано**. КЛА має вигляд багатополіосника з m вхідними і n вихідними клемами. Припустимо, що до складу множин X_i і Y_j включені і фіктивні елементи 0 , наявність яких на вході або виході КЛА означає відсутність сигналу на відповідній клемі.

Отже, вхідний сигнал на КЛА має вигляд $x = (x_1, x_2, \dots, x_n)$, а вихідний $y = (y_1, y_2, \dots, y_m)$.

Для програмної реалізації агрегативних моделей **даними** вважаються: елементарні дані, списки даних, масиви даних і структури даних.

Агрегативні системи служать певним узагальненням таких схем як автомати і моделі масового обслуговування, мережі Петрі, чисельні методи рішень диференціальних рівнянь, моделі «чорного ящика». Структури даних,

що описують стани і сигнали агрегативних систем, допомагають формалізувати концептуальне подання, які є у користувача щодо елементів складної системи.

В основу поняття агрегативної моделі покладене структурне подання системи у вигляді взаємодіючих елементів – КЛА. Динаміка агрегативної системи повністю визначається послідовністю подій, що відбуваються в ній. Це відповідає концепції алгоритмічної моделі динамічної складної системи. Агрегативні системи зручно реалізуються на комп'ютері.

12.4.6 Мережі Петрі

Мережі Петрі і їх узагальнення являють собою математичні моделі, побудовані в рамках певної концепції структуризації. Вона базується на можливості подання систем у вигляді сукупності паралельних процесів, взаємодіючих на основі синхронізації подій або розподілу для декількох процесів загальних ресурсів. Кожний процес в рамках цієї концепції представляється у вигляді логічно обумовлених не впорядкованих за часом причинно-наслідкових ланцюжків умов і подій.

При розробці структур моделей дискретних систем можна використовувати дані про логічний взаємозв'язок спостережуваних в системі подій і умов, що визначають настання цих подій, як базову інформацію. В деяких реальних системах не можна точно вказати момент часу настання подій. Його випереджає складна система причин і слідств. Точне знання моментів часу реалізації подій в системі часто можна ігнорувати, оскільки такі відомості про події, що відбуваються в реальних або проєктованих системах, або просто відсутні, або їх не можна вважати достовірними. Це пояснюється різноманіттям подій, що випереджають, умов, неможливістю повного їх урахування і вірного опису, а також дією складної і заплутаної системи причин і слідств, визначення яких на часовій шкалі часто не представляється можливим.

Вводяться базові поняття “Умова” і “Подія”, які можуть бути зв'язані відношенням типу “Виконується після”. Події виражають дії, реалізація яких управляє станами системи. Стани задаються у вигляді складних умов. Вони

формулюються як предикати з перемінними у вигляді простих умов. Тільки при досягненні певних станів (в цьому випадку відповідні предикати приймають істинне значення) забезпечується можливість дій (настання подій). Умови, з фактами виконання яких зв'язана істинність предиката і, отже, можливість реалізації події, називаються «до-умовними».

Подальшим розширенням розфарбованих мереж з'явилися **предикатні мережі**. Вони дозволили зв'язати з переходами мереж логічні формули (предикати або захист), що представляють класи можливих розміток у вхідних і вихідних позиціях відповідно до міток дуг. Ці вираження задають умови відбору необхідних фарб для спрацьовування переходів.

При розробці імітаційної моделі з класу розширень мереж Петрі, як і для будь-якої іншої імітаційної моделі, виділяються чотири основні етапи: структуризації, формалізації і алгоритмізації, програмування моделі, а далі проведення імітаційних експериментів з моделлю.

При структуризації визначаються і неодноразово уточнюються діючі в системі процеси і використовувані ресурси, множини позицій, які відображають у моделі стани процесів і ресурсів, і множини переходів (подій), підмножини синхронізуючих переходів (для опису паралельних процесів).

При формалізації й алгоритмізації елементів моделі для кожної позиції визначаються атрибути. Перехід вважається формально описаним, якщо відомі множини суміжних із цим переходом позицій, умови ініціації переходу, схема виконання, процедура переходу.

Умови ініціації переходу – є деякий предикат, що приймає істинне значення, якщо реалізується деяка розмітка позицій множини (перевіряються атрибути міток). Схема виконання визначає зміну розмітки позицій мережі при спрацьовуванні переходу. Процедура переходу являє собою правила обчислення атрибутів або додавання міток.

Програмування моделі пов'язане з описом позицій і переходів мережі, оформлюваних за допомогою деяких мов програмування або моделювання, наприклад, GPSS.

12.5 Моделі системної динаміки

12.5.1 Загальні відомості про системну динаміку

Моделі системної динаміки мають діло з безперервними системами, стан яких змінюється залежно від деяких незалежних перемінних (звичайно часу). Мови імітаційного моделювання безперервних систем призначені для моделювання динамічних об'єктів з безперервним фазовим простором і безперервним часом. Як правило, такі об'єкти описуються за допомогою систем диференціальних і інтегро-диференціальних рівнянь.

Класичним прикладом мов моделювання такого типу є мова **DYNAMO**, яка була запропонована Дж. Форрестером. Розроблені їм концепція системної динаміки і потоковий підхід в імітації викладені в його основних працях [44-46], де досліджуються динаміки підприємств, міст, розвитку людської цивілізації.

Методи і техніка побудови моделей системної динаміки дуже вплинули на формування технології імітаційного моделювання. Зараз з'явилося багато систем моделювання, таких як **IThink, Vensim, Powersim**, в яких можливості моделювання безперервних і нелінійних динамічних систем доповнені зручними графічними інтерфейсами. Багато компаній пропонують консалтингові послуги з моделювання систем на основі інструментальних засобів, які підтримують нормативні підходи методів системної динаміки. Існує міжнародне наукове товариство, яке розвиває і удосконалює ці методи.

Моделі системної динаміки одержали широке поширення при дослідженні складних систем сфери виробництва й економіки, торгівлі і міського господарства. З їх допомогою вирішують соціальні й екологічні проблеми [25, 44-46]. Ці моделі поклали початок новому напрямку в системних дослідженнях – **глобальному моделюванню**, яке охоплює проблеми світової економічної системи і світового розвитку [45].

Концепція системної динаміки орієнтована на моделювання систем і процесів на високому рівні агрегування, де відображення окремих елементів процесів стає непотрібним. Наприклад, це – моделювання економіки окремих

країн і регіонів, транспортних систем країни, і т.п. В основі концепції системної динаміки лежить представлення функціонування системи як сукупності потоків інформації, енергії, промислової продукції, коштів і т.п.

12.5.2 Структура моделей системної динаміки

Математичною основою методів системної динаміки є системи диференціальних рівнянь, які описують динамічні процеси в просторі станів:

$$\dot{X} = f(x, u, t), \quad (12.1)$$

де $x = (x_1, x_2, \dots, x_m)^T$ – вектор станів;

x_1, \dots, x_m – перемінні станів;

$u = (u_1, \dots, u_p)^T$ – вектор **входів**;

$t \in T$ – час.

Диференціальні моделі, які застосовуються в математичній теорії систем, включають крім рівнянь (12.1), ще і рівняння вигляду:

$$y = H(x, u), \quad (12.2)$$

в яких перемінна $y = (y_1, \dots, y_q)^T$ – вектор **виходів** процесів (час t тут для стислості опускається).

При складанні диференціальних моделей провадиться вибір перемінних станів і встановлюються зв'язки між ними у вигляді функцій правих частин рівнянь стану. Сформулювати такі залежності тільки з використанням перемінних стану буває, як правило, дуже важко. Більш продуктивним виявляється підхід, заснований на детальному описі ланцюжків причинно-наслідкових зв'язків між факторами, відображуваними в моделі за допомогою перемінних стану.

При побудові моделей системної динаміки відомості про об'єкти дослідження від експертів піддаються ретельному аналізу, в ході якого провадиться вибір і інтерпретація перемінних стану моделі, виявляються і описуються причинно-наслідкові відносини між перемінними стану цих відносин у формі структурованих функціональних залежностей.

При опису вектор-функцій $f(x, u)$ в ролі загальної структурної схеми використовується граф, вершинам якого відповідають перемінні моделі, а дугам – безпосередні функціональні зв'язки між цими перемінними.

Приклад. Нехай потрібно описати структуру правих частин рівнянь стану: $x_1 = f_1(x_1, x_2, u_1)$; $x_2 = f_2(x_1, x_2, u_1)$. Після факторизації правих частин цих рівнянь можуть бути отримані структуруючі співвідношення вигляду:

$$f_1(x_1, x_2, u_1) = F_1(w_1, w_2); w_1 = F_2(x_1, x_2); w_2 = F_3(w_1, u_1);$$

$$f_2(x_1, x_2, u_1) = F_4(w_2, u_1).$$

Цьому випадку буде відповідати граф безпосередніх функціональних залежностей між перемінними диференціальної моделі, представлений на **рис.12.1**.

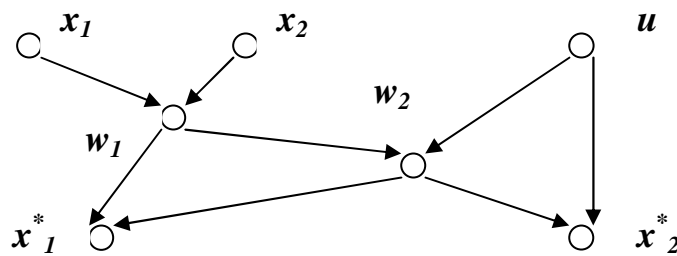


Рис.12.1 Граф функціональних залежностей між перемінними

Опис структури даного графу представлено в **табл. 12.1**.

Таблиця 12.1

Опис структури графу

Вершини графу	Яруси графу
x_1, x_2, u_1	0
w_1	1
w_2	2
x_1, x_2	3

Структури правих частин рівнянь диференціальних моделей завжди можуть бути представлені багатоярусними графами, що не мають контурів. При розробці таких ланцюжків причинно-наслідкових зв'язків було необхідно ввести деякі **додаткові** перемінні. Компоненти векторної перемінної $w = (w_1, \dots, w_s)^T$ створять в сукупності множину додаткових перемінних, що включаються при структуризації правих частин рівнянь стану (12.1).

Загальна схема побудови структур правих частин рівнянь стану цих моделей може бути представлена у вигляді графа, нульовий ярус якого утворюють вершини, що відображають на графі перемінні стану й вхідні перемінні моделі. Останній ярус становлять вершини, що представляють перемінні лівих частин рівнянь стану (12.1). Вершинам всіх проміжних ярусів відповідають перемінні стану і додаткові перемінні диференціальної моделі.

Рівняння стану моделей системної динаміки складаються у формі:

$$X_{t+1} = X_t + hf(x_t, u_t),$$

де h – крок дискретизації, $t = 0, 1, 2, \dots$

В загальній структурі моделей системної динаміки виділяється мережа потоків і мережа інформації. В мережі потоків перемінні стану і перемінні швидкості зміни стану (відповідно, рівні і темпи в термінах системної динаміки), що визначають стан моделі, задаються системами різностних рівнянь і в описі присутні в неявному виді. За допомогою мережі інформації структурується функція $f(x_t, u_t)$.

12.5.3 Представлення потоків

Модель представляється у вигляді мережі потоків матеріальних інгредієнтів моделі. Кожний компонент цієї мережі відповідає якійсь одній сукупності однорідних інгредієнтів, динаміка яких ураховується в моделі. На підприємстві, наприклад, можна виділити фінансові, матеріальні, людські потоки, та ін. Мережа має вузли і дуги. Вузли компонентів, за винятком нульового вузла, зображують найбільш істотні з погляду розроблювачів моделі стану виділених інгредієнтів, а дуги задають можливі переходи їх елементів з одного стану в інші.

Розподіл елементів по станах міняється з часом. Ці зміни для системної динаміки є нормативними образами процесів, що моделюються. Перемінні x_1, \dots, x_m рівнянь стану моделі розглядаються, як характеристики розподілу елементів, які входять в модель інгредієнтів по станах X_1, \dots, X_m .

Перемінні v_1, \dots, v_n приймаються за характеристики швидкостей, з якими відбуваються переходи елементів з стану в стан за дугами V_1, \dots, V_n мережі.

Вузли мережі зображуються у вигляді прямокутників. Нуль мережі прийнято позначати спеціальним знаком. Така інтерпретація нагадує структурні форми завдання автоматних моделей дискретних процесів.

Однак моделі системної динаміки – це диференціальні моделі, перемінні стани яких безперервні. Тому наведеної тут структурної концепції рівнянь стану моделей системної динаміки звичайно дається більш природне пояснення, засноване на гідравлічній інтерпретації потокових мереж. В цій інтерпретації кожний вузол $x \in X$ мережі розглядається як резервуар, рівень наповнення якого в момент t дорівнює $x_i(t)$, $t=1,2,\dots, m$. Дуги $v \in V$ мережі відповідають потокам рідини між резервуарами. Вони вказують напрямки потоків, темпи яких характеризуються об'ємними видатками рідини $v_j(t)$, $j = 1, 2, \dots, n$ на інтервалі часу $[th, (t+1)h]$. Темпи потоків символічно зображуються у вигляді вентилів на дугах потокових мереж.

Наочність гідравлічної інтерпретації обумовила її важливу роль при структуризації моделей, спеціальну “потокову” термінологію і загальне трактування математичних схем моделей системної динаміки. Відповідно до цієї термінології перемінні стани x_1, \dots, x_m називаються **рівнями** моделі, а перемінні v_1, \dots, v_n – **темпами**. Рівні і темпи – основні перемінні моделей. Всі інші перемінні називаються допоміжними. Вони використовуються при структуризації функціональних залежностей f темпів від рівнів і входів, а також функціональних залежностей H виходів y від рівнів x і входів u .

Таким чином, при опису динамічної системи процеси відображаються у вигляді деякої фіксованої структури, яка складається з **накопичувачів-рівнів**, з'єднаних взаємозалежними **потоками**. Вони, перетікаючи по всій системі, змінюють значення рівнів. Рівні характеризують виникаюче нагромадження всередині системи і є величинами, які визначаються, як перемінні стану системи. Рівні описують величини, безперервні по діапазону значень, але дискретні в часі. Наприклад, для банку - це сальдо, для складу - поточний рівень запасів на складі.

Значення рівнів змінюють потоки. Швидкості зміни потоків, тобто темпи, регулюються в економіці управлінськими рішеннями. Темп показує, як

змінюються рівні за інтервал часу, рівний кроку моделювання. Темп характеризує динаміку системи, що моделюється. Якщо систему зупинити, то рівні будуть значимі, а темпи буде неможливо розрізнити.

Потокові мережі є неявною формою опису станів системи (перемінних стану і швидкостей зміни станів), у формі різностних рівнянь. Основні перемінні моделі: рівні, темпи, допоміжні описуються за допомогою наступного рівняння:

$$X(t+h) = x(t) + h \times V(t), \quad (12.3)$$

де t – модельний (системний) час;

h – крок моделювання (інтегрування) – зміна (збільшення) часу;

$x(t), x(t+h)$ – значення рівня в моменти часу;

$V(t)$ – темп зміни рівня, тобто величина його зміни в одиницю часу.

Рівні імітують характеристики системи, які визначають її стан у конкретний момент часу. Закон зміни темпу задається функціональною залежністю:

$$V(t) = F(p_1(t), p_2(t), \dots, p_k(t)), \quad (12.4)$$

де F – довільна функція від k аргументів;

$p_i(t)$ – будь-які перемінні, значення яких у момент t відомі.

Таким чином, алгоритм імітації реалізується у такий спосіб:

Встановлюються параметри системного часу (початкове значення, крок інтегрування, довжина інтервалу моделювання) і задаються початкові умови – значення рівнів у початковий момент часу.

Розраховуються значення всіх темпів і допоміжних змінних на даний момент системного часу t , який потім збільшується на крок моделювання (інтегрування) $t+h$. За рівняннями (12.3) розраховуються значення всіх рівнів на даний момент системного часу. Далі виконуються ітерації за цією схемою, поки не пройде весь інтервал моделювання.

12.5.4 Інформаційна мережа

Структуризація функціональних залежностей f (12.1) і H (12.2) завершується побудовою інформаційної мережі моделі. Їх описують за допомогою багатоярусних графів з використанням заданих рівнянь моделі. Для складних

слабоструктурованих систем такі рівняння, як правило, не відомі. Тому їх моделі створюються шляхом структуризації експертної інформації. При цьому аналізуються ланцюжки причинно-наслідкових зв'язків факторів, які треба, на думку експертів, обов'язково враховувати при опису динаміки станів об'єкту, який моделюється. Допоміжні перемінні головним чином використовуються для побудови логічно ясних, добре інтерпретованих структур взаємозв'язків перемінних, за допомогою яких у моделях відображаються різноманітні відомості про об'єкт моделювання, які надаються експертами.

12.5.5 Діаграми причинно-наслідкових зв'язків.

Діаграми причинно-наслідкових зв'язків є попереднім способом аналізу складної системи. Вони забезпечують опис **якості** взаємозв'язків факторів, урахування яких визнається експертами необхідним для відображення в моделі принципів моментів розвитку процесів, що моделюються.

На дугах графів причинно-наслідкових діаграм розставляються знаки “+” і “-”, за допомогою яких фіксується передбачуваний або ж емпірично обґрунтований характер впливу перемінних, відповідних кінцевим вершинам дуг. Наявність знака “+” на дузі (*a*, *b*), спрямованої з вершини *a* графа до вершини *b*, означає, що ріст перемінної *a* приводить до росту перемінної *b*. Знак “-” описує протилежний ефект впливу *a* на *b*: з ростом *a* величина *b* убуває. Граф із проставленими вагами на дугах називається зваженим оргграфом, а з указаною функцією – функціональним оргграфом.

Технологія складання діаграми причинно-наслідкових зв'язків наступна. На основі вербального опису процесів, що моделюються, виділяють фазові перемінні. Використовуючи логіку опису, їх попарно класифікують за критерієм “причина-наслідок”. Причина з наслідком з'єднується стрілками. Відповідно до впливаючих факторів, над стрілкою ставляться знаки «плюс» або «мінус». В результаті такої процедури позначаються і контури зі зворотним зв'язком. Така схема структуризації інформації про причинно-наслідкові взаємозв'язки динамічних процесів в об'єктах дозволяє одержати загальну структуру диференціальної моделі системної динаміки.

12.5.6 Системні потокові діаграми моделей

Мережі потоків і ярусні інформаційні мережі описують структуру рівнянь моделей системної динаміки тільки в роздріб. Причинно-наслідкові діаграми, навпаки, відображають взаємозв'язки перемінних моделей тільки в цілому, не розділяючи їх за типами. Мережа потоків є неявною формою (у вигляді різницевих рівнянь), а мережа інформації – явною формою опису тих самих перемінних.

При розробці диференціальних моделей системної динаміки використовують техніку графічного опису структур систем, що моделюються, у вигляді **системних поточкових діаграм**. Вони поєднують графи мереж потоків і інформації. В результаті цього забезпечується цілісність представлення структури рівнянь (12.1), (12.2). В поточковій діаграмі використовуються нормативні структури поточкових й інформаційних мереж. При використанні спеціальної символіки графи перетворюють поточкові діаграми в засоби наочного відображення інформації про динаміку процесів, що моделюються, і в мову спілкування експертів і системних аналітиків.

Об'єднання в конструкції поточкових діаграм форм явного (мережа інформації) і неявного (мережа потоків) опису залежностей перемінних стану модельованих систем, а також розвинена графічна символіка приводять до того, що поточкові діаграми дають значну частину тієї ж інформації, що і системи рівнянь моделі, але в більш наочній формі.

Побудова поточкових діаграм безпосередньо пов'язана з рішенням завдань перед модельного аналізу досліджуваної проблеми, служить свого роду підсумком вихідного змістовного пророблення інформаційної бази процесу моделювання і забезпечує ефективну структурування знань експерта.

Системні поточкові діаграми дозволяють здійснювати декомпозицію складної системи з наступною композицією. Така мова визначає форму вираження обговорюваних питань, виступає як засіб поділу на частки завдань аналізу причинно-наслідкової структури системи, що моделюється, і наступного “складання” їхніх результатів у цілісну картину організації процесів функціонування і розвитку системи.

12.5.7 Основні етапи технології системної динаміки

У концепції системної динаміки є дві **нормативні схеми** формування загальної структури моделей, які застосовуються залежно від класу розв'язуваних завдань.

Схема 1. Спочатку розробляється причинно-наслідкова діаграма моделі. При її розробці в число факторів, що враховуються, і зв'язків включаються всі ті, які використовуються експертами при змістовному описі об'єкту, що моделюється. Потім виконується аналіз зафіксованих у розробленій діаграмі ланцюжків причинно-наслідкових зв'язків і визначаються фактори, які описуються в моделі **рівнями і темпами**, тобто виділяються перемінні рівнів і темпів. У результаті формується на ескізах **мережа потоків моделі**. Далі виділяється й уточнюється **інформаційна мережа** моделі як структура, що доповнює мережу потоків у причинно-наслідковій діаграмі.

Схема 2. Спочатку виділяється множина основних матеріальних інгредієнтів, динаміку яких необхідно відобразити в моделі. Для кожної виділеної сукупності однорідних елементів визначається множина їхніх можливих станів і встановлюється структура переходів елементів інгредієнтів зі стану в стан. В результаті формується мережа потоків моделі. Після цього визначається структура причинно-наслідкових зв'язків між рівнями й темпами мережі потоків, тобто розробляється структура інформаційної мережі моделі. При такому підході за допомогою інформаційної мережі “як би” зшиваються потокові представлення.

При нормативному підході до розробки динамічних моделей виділяються наступні основні етапи технології системної динаміки:

Етап 1. Концептуалізація проблемної ситуації, результатом якої є вербальна модель і знаковий оргграф.

Етап 2. Побудова системних поточкових діаграм.

Етап 3. Параметризація моделі.

Етап 4. Формалізація, результатом якої є машинна модель.

На **етапі 1** в процесі побудови вербальної моделі здійснюються постановка проблеми, аналіз вихідної інформації, формулювання цілей

моделювання. Успіх виконання цього відповідального і складного етапу залежить від рівня підготовки та досвіду системних аналітиків, які створюють моделі. При складанні вербального опису систематизуються причинно-наслідкові описи динамічних процесів, що моделюються. Він може містити ескізи потокових діаграм і діаграми причинно-наслідкових зв'язків. Для складання вербального опису необхідно:

- проаналізувати вихідну інформацію, що виявляється в ході дискусій;
- ухвалити проблеми і сформулювати цілі моделювання;
- сформулювати принципові гіпотези, які мають знайти відбиття в моделі;
- виявити границі системи, що моделюється;
- детально встановити і обґрунтувати склад взаємодіючих компонентів, динаміка яких визначає істотні аспекти поведження і стану системи;
- обговорити впливи на систему зовнішніх факторів;
- виявити основні фактори і процеси, відображення яких є обов'язковим для досягнення поставленої мети моделювання;
- описати всі структури причинно-наслідкових взаємозв'язків між факторами, відображуваними у моделі.

При розробці вербальної моделі повинні бути виявлені основні ситуації, варіанти стратегій, експериментальне дослідження яких передбачається проводити за допомогою імітаційної моделі, критерії оцінки поведження моделі, часові параметри імітації (крок інтегрування, час моделювання).

В результаті виконання першого етапу вербальний опис повинен містити попередній опис всієї структури причинних взаємозв'язків у вигляді ескізів діаграм потоків і/або діаграми причинно-наслідкових зв'язків.

На **етапі 2**, який є основним етапом при структуризації моделі системної динаміки, виконується перехід від причинно-наслідкової діаграми до побудови **системних потокових діаграм**. При використанні схеми 1 нормативного підходу виділяються вершини і дуги орграфу причинно-наслідкової діаграми відповідно до основних типів перемінних і аксіом системної динаміки. Виконання аксіом забезпечує надалі розробку й алгоритмізацію диференціальних (різностних) рівнянь моделі. В цілому цей перехід неформальний і, як правило, опирається на змістовні судження про характер і причини взаємодії факторів, представлених у вигляді перемінних моделі.

Параметризація моделі (етап 3) являє собою процес перекладу вербальних описів взаємозалежностей факторів ситуації, що моделюється, на мову чітких кількісних співвідношень. В системній динаміці використовуються два основних прийоми вибору й обґрунтування виробляючих функцій темпів. В першому з них темпи потоків розглядаються як функції прийняття рішень. Застосування цього прийому є зручним і природним при моделюванні виробничих і економічних систем, коли виробляючі функції темпів фактично являють собою кількісні описи вирішальних правил, що діють у механізмі керування системою.

Рекомендується виділяти і відображати у виробляючих функціях темпів наступні структурні елементи:

- бажаний стан потокового сектора, в якому діє певний темп (завдання мети вирішального правила);
- існуючий (поточний) стан сектора;
- кількісне вираження розходження між зазначеними станами потокового сектора;
- співвідношення для вироблення коригувального впливу на темп, що забезпечує переведення сектора в бажаний стан.

Наприклад, у виробничій системі може мати місце неузгодженість між мережею матеріалів і мережею устаткування, що пов'язано з відхиленнями від необхідної економічної програми. З огляду на це, треба в інформаційній мережі моделі формувати відповідні регулятори для досягнення необхідного стану виробничої системи.

Другий методичний прийом – це завдання виробляючої функції темпу у вигляді добутку “нормального темпу” і коригувальних множників, що визначають його залежність від перемінних стану (рівнів) моделі.

На **рис. 12.2** наведено фрагмент потокової діаграми моделі світової динаміки Дж.Форрестера [45].

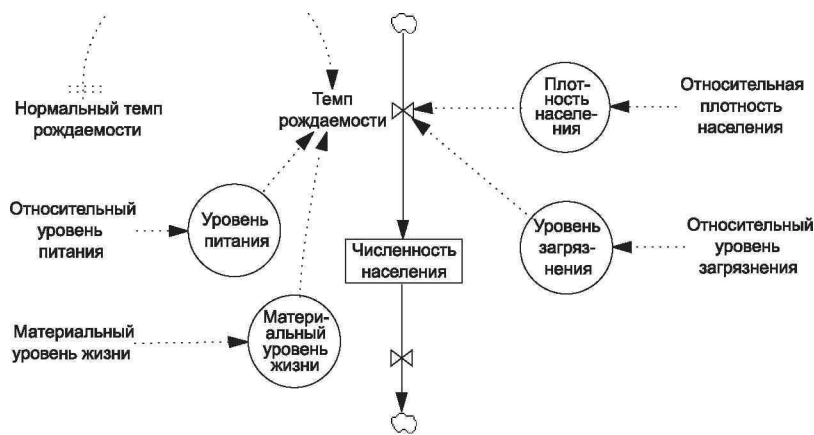


Рис.12.2 Фрагмент потокової діаграми моделі світової динаміки

Темп народжуваності населення визначається тут як добуток чисельності населення (стан), нормального темпу народжуваності і співмножників, що відображають залежності темпу народжуваності від матеріального рівня життя, густоти населення, рівня харчування і рівня забруднення. Кожний з цих співмножників являє собою нелінійну функцію, вигляд якої відображає реальні сукупності даних про характер описуваного причинного зв'язку або задає експертну оцінку (гіпотезу) такого зв'язку.

Описана форма рівнянь темпів забезпечує наочність і простоту їхньої змістовної інтерпретації і є зручними засобом для експертної оцінки інформації для аспектів, що найбільш важко формалізуються, причинних зв'язків компонентів процесів. Необхідно, щоб структури рівнянь темпів чітко відповідали побудові інформаційної мережі потокової діаграми моделі системної динаміки. Наприклад, кожний з коригувальних множників ставиться у відповідність дузі інформаційної мережі потокової діаграми моделі. При цьому вигляд функції (зростаюча або убутна) буде відповідати знаку, певному для даної дуги в причинно-наслідковому графі моделі.

Таким чином, використовуючи другий з розглянутих прийомів, фахівець із системної динаміки на основі знакового орграфу моделі може записати загальний вираз для будь-якого темпу моделі й якісно охарактеризувати вигляд утворюючих його функціональних співмножників.

Сьогодні існує велика кількість систем моделювання, таких як **Vensim**, **IThink**, **Powersim** та ін., що мають ефективні інструментальні засоби програмної підтримки техніки, процедур і методів системної динаміки. З їхньою допомогою системні потокові діаграми створюються на ідеографічному рівні, параметризація моделі здійснюється в режимі меню, з використанням засобів введення формульної, табличної і графічної інформації у процесі діалогової взаємодії розроблювачів моделі й системи моделювання. Вони забезпечують прийом специфікацій стандартних описів, що задаються експертами, їхній контроль на несуперечність, перетворення отриманої інформації в текст мовою моделювання.

Такі системи моделювання мають розвинені засоби для аналізу результатів обчислювальних експериментів і проведення сценарних розрахунків. Знакові графи використовуються при моделюванні сценаріїв розвитку макроекономічних процесів, виборі варіантів економічної політики. Разом з балансовими методами моделі системної динаміки знаходять широке застосування в моделюванні соціально-економічних процесів, у моделях ресурсного типу, при дослідженні процесів відтворення в регіональних і макроекономічних системах [11, 18, 25]. За допомогою цих методів досліджуються слабоструктуровані задачі.

Відсутність теоретичних знань, якісний характер знань про систему з великою часткою експертних знань не дозволяє застосовувати точні нормативні моделі. При дослідженні такого класу систем присутній низький рівень точності вихідних даних, зовнішня і внутрішня невизначеність, пов'язана з наявністю великої кількості факторів, що перебувають під слабким контролем осіб, які приймають рішення.

Особливістю динамічного моделювання є те, що рішення тут носять якісний характер. Його результати дозволяють в основному судити про напрямок розвитку динамічних процесів, проводити аналіз стабільності динамічних процесів.

12.6 Імітаційне моделювання в системі ARIS

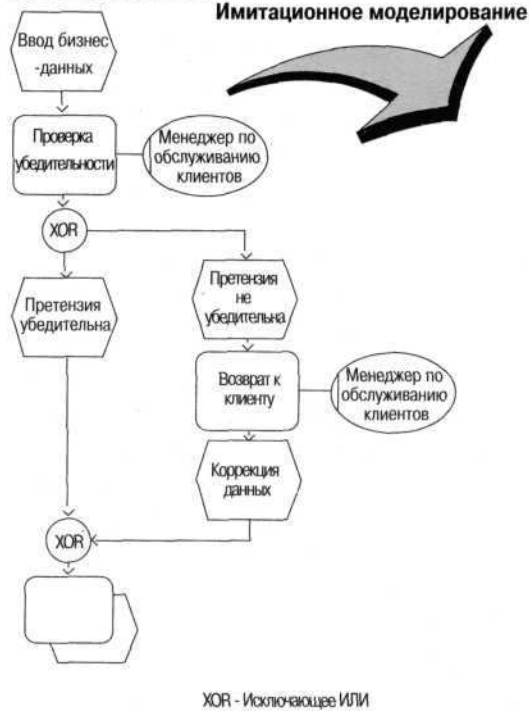
Для оптимізації бізнес-процесу розробляється декілька його альтернатив, динаміка яких досліджується за допомогою методу імітаційного моделювання. Окремі процеси генеруються і відслідковуються на моделі бізнес-процесу, побудованій з використанням засобів, описаних у розділі 8 тому 1. Процеси описуються на рівні екземплярів і аналізуються їхні взаємини. Це дозволяє виявити потенційні затримки до початку виконання реального процесу.

Альтернативи виробляються індивідуально на основі емпіричних досліджень або автоматичним способом - за випадковим принципом. Вони можуть відрізнитися за такими аспектами, як структура процесу, час виконання функцій, характер поведінки відповідних організаційних одиниць.

Структура імітаційної моделі формується безпосередньо із загальної структури процесу, а потім оцінюється за допомогою модуля імітаційного моделювання. На **рис.12.3** наведено приклад оцінки бізнес-процесу за допомогою системи SIMPLE++ разом з системою ARIS Toolset, яка описана у тому 1 (розділ 6).

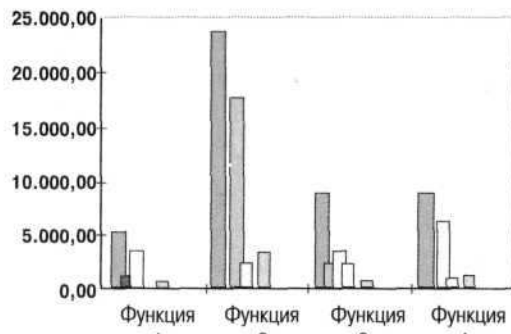
Подібні імітаційні експерименти для розробки оптимальних бізнес-процесів застосовуються для організації виробництва, наприклад, при складанні планів розміщення устаткування на промислових підприємствах, а також для визначення ефективних рішень при управлінні процесами. Для процесів, що відбуваються в офісах, такі дослідження проводяться рідше, хоча у зв'язку з оптимізацією процесів управління й обслуговування вони починають набувати все більшої популярності. Прикладом тут є моделювання банківської і страхової діяльності.

**Событийная диаграмма
процесса (EPC)
Обработка претензий**

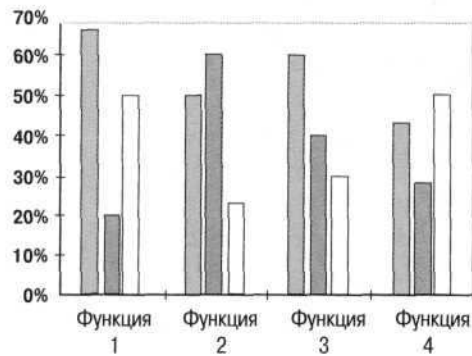


Результаты имитационного моделирования:

Общая характеристика стоимости



Общая характеристика мощностей



- Общая стоимость [Руб.]
 - Стоимость материалов [Руб.]
 - Стоимость рабочей силы [Руб.]
 - Вспомогательные и эксплуатационные расходы [Руб.]
 - Стоимость энергоносителей [Руб.]
 - Различные накладные расходы [Руб.]
 - Ставки списания [Руб.]
 - Вмененный процент [Руб.]
 - Прочие издержки
-
- Людские ресурсы
 - Ресурсы ИТ
 - Прочие ресурсы

Рис.12.3 Пример оцінки бізнес-процесу при імітаційному моделюванні

Розділ 13

Пакети програм імітаційного моделювання

13.1 Короткий огляд створення і розвитку пакетів програм імітаційних систем

В [64] імітаційне моделювання визначається як різновид аналогового моделювання, реалізованого за допомогою набору математичних інструментальних засобів, спеціальних комп'ютерних програм, що імітують, і технологій програмування, що дозволяють за допомогою процесів-аналогів провести в пам'яті комп'ютера цілеспрямоване дослідження структури і функцій реального складного процесу в режимі «імітації» і виконати оптимізацію деяких його параметрів.

За допомогою імітаційного моделювання і відповідних сучасних програмних засобів можна створювати ті моделі, які неможливо сформулювати традиційними методами.

Застосування імітаційного моделювання є виправданим, якщо питання, відповідь на які повинна дати модель, ставляться не до з'ясування фундаментальних законів і причин, що визначають динаміку реальної системи, а до практичного аналізу поведінки системи [8-10,35].

Одним з перших методи імітаційного моделювання для аналізу економічних процесів застосував Т. Нейлон [28]. Протягом 1970-80-х років спроби використовувати цей вид моделювання в реальному управлінні економічними процесами носили епізодичний характер через складність формалізації економічних процесів і недостатню розвиненість програмних засобів. Однак саме в середині 1970-х років з'явилися перші інструментальні засоби імітаційного моделювання, наприклад, система **GPSS**, що дозволяла створювати моделі контрольованих процесів і об'єктів в основному технічного або технологічного призначення [10].

В 1980-х роках системи імітаційного моделювання стали розвиватися більш активно. В той час у різних країнах миру застосовувалося понад 20 різних систем (**GASP-IV**, **SIMYLA-67**, **SLAM-II**, **GPSS-V** і т.д.).

В 1990-х роках з'явилася множина пакетів для імітаційного моделювання, які ставали все більш розвиненими і зручними навіть для непідготовленого користувача. Це можна пояснити не тільки збільшенням інтересу до імітаційного моделювання, але й більшим прогресом у розвитку комп'ютерної техніки, швидкодія і можливості якої зростають з кожним днем. Можна виділити такі засоби, як **Powersim, Ithink, Process Charter, Simula**.

Існує два підходи до імітаційного моделювання - **статичний і динамічний**. Статичні моделі являють собою системи рівнянь, які вирішуються один раз. Динамічні моделі містять у собі ще одну перемінну – час. Вони бувають двох типів: **безперервні і дискретні**. У безперервних моделях час змінюється лінійно, а процес - у безпосередній залежності від часу. За їх допомогою можна, наприклад, моделювати рух потоку рідини в трубопроводі. В дискретних моделях перемінними є події і часові інтервали. Такі моделі, наприклад, дають можливість управляти складськими запасами. Виконуючи розрахунки параметрів бізнес-систем на різних часових інтервалах, розроблювач моделі може досліджувати розвиток систем у часі. Для моделювання бізнес-систем в основному застосовуються пакети динамічного моделювання. Характеристика найбільш використовуваних з них [10,65] представлена в **табл. 13.1**. В пакеті **Process Charter** модель будується за допомогою блок-схеми. Пакети **Powersim** і **Ithink** використовують систему позначень **Systems Dynamics**, запропоновану в 1961 р. Джейм Форрестером з Масачусетського технологічного інституту США. Пакети **Extend** і **Pilgrim** застосовують компоновочні блоки, а пакет **Vensim** – потокові діаграми.

Підтримка аналізу чутливості означає можливість багаторазового виконання моделі з різними вхідними параметрами і порівняння результатів.

Таблиця 13.1

Порівняння пакетів імітаційного моделювання

Пакет	Виробник	Напрямок	Графіка	Анімація	Підтримка
Process Charter 1.0.2	Scitor Менло-Парк, США	Дискретне моделювання	Блок-схеми	–	–
Powersim 2.01	Modell Data AS, Берген, Норвегія	Безперервне моделювання	Потокові діаграми	–	–
Ithink 3.0.61	High Performance System, Inc Ганновер, США	Управління фінансовими потоками, реінжиніринг підприємств, банків і т.і.	CASE-Кошти, потокові діаграми	+	Аналіз чутливості
Extend+ BPR 3.1	Imagine That, Inc. Сан-хосе, США	Стратегічне планування, бізнес-моделювання	Компоновочні блоки, безперервні і дискретні моделі	+ мова Modl	Аналіз чутливості
Pilgrim	МЕСИ, Росія	Моделі динамічних систем	Компоновочні блоки, CASE-засоби	Мова. 3++	+
Vensim	Ventana Systems	Моделі системної динаміки	Потокові діаграми	–	+

13.2 Опис пакетів імітаційного моделювання

13.2.1 Пакет Process Charter

Компанія Scitor назвала пакет **Process Charter** інтелектуальним засобом створення блок-схем. Це спрощує розробку базової моделі, оскільки більшість користувачів знайомі з принципами побудови блок-схем. Цей пакет менш ефективний у порівнянні з іншими пакетами, що містять більш розвинені засоби імітаційного моделювання для побудови безперервних моделей і вихідних діаграм. Але він дозволяє успішно проводити дискретне моделювання в тих випадках, коли потрібен аналіз вартості.

Компанія Scitor включила у свій продукт множину типів блоків, стилів рядків і видів тексту для форматування діаграм моделей. Наявність різних засобів форматування, таких як заголовки, півтони, кольори, жирний шрифт і інструментальні лінійки з більшою кількістю кнопок, робить **Process Charter** схожим на будь-який інший **Windows**-Пакет побудови блок-схем.

Кожний із блоків, що одержав назву **Activity** (дія), має ряд атрибутів, спеціально призначених для моделювання. Перший блок управляє потоком елементів, що проходять через систему. Така організація вказує на те, що пакет **Process Charter** в основному орієнтований на дискретне моделювання. Однак завдяки тому, що моделі, створені за його допомогою, змінюються в часі з постійною швидкістю, його можна настроїти також на безперервне моделювання.

Побудувавши блок-схему, що описує фізичну основу моделі, розроблювач, використовуючи електронну таблицю **Resource**, визначає ресурси, необхідні для моделі. Наприклад, при моделюванні гіпотетичного центра обслуговування можна задати три черги і виділити кожному з них обслуговуючого оператора, як одного з ресурсів. При визначенні ресурсів необхідно вказати їхній тип (працівник або матеріал) і характеристику (розхідний або багаторазово використовуємий), а також ввести атрибути вартості й кількості. Наявність атрибута вартості робить пакет **Process Charter** підходящим засобом для аналізу витрат, а наявність атрибута кількості дозволяє вручну збільшувати число операторів і проводити дослідження чутливості, багаторазово запускаючи модель.

Однак, оскільки пакет **Process Charter** позбавлений спеціальних елементів керування, він не дає можливості будувати авторські моделі. Процес присвоєння ресурсів діям організований за принципом, відповідно до якого представляються зв'язки між діями. Це дозволяє розроблювачу вказати атрибути для кожної з них.

Під час виконання моделі проходження ресурсів по блоках дій показується на екрані за допомогою кольорової анімації. Це дає можливість в оперативному режимі проводити попередній аналіз процесу. Щоб переглянути вихідну інформацію, розроблювач може звернутися або до сторінки результатів у блокноті дій, або до однієї з п'яти електронних таблиць. Найбільш важливі вихідні

значення можна помістити в електронну таблицю **Key Values** для колективного перегляду. Крім того, пакет містить 13 заздалегідь сформованих гістограм для перегляду інформації про моделювання. Однак, в пакеті не передбачені засоби побудови власних гістограм і застосовувати різні типи діаграм, такі як лінійні, кругові і діаграми розкиду, а також включати в гістограми інші дані.

Недоліками пакета **Process Charter** є невелика кількість можливостей, слабка підтримка моделювання безперервних компонентів, обмежений набір засобів для аналізу чутливості й побудови діаграм.

13.2.2 Пакет Powersim

Пакет **Powersim** є засобом створення безперервних моделей, але він малоефективний для дискретного моделювання, що є його недоліком. Для використання цього пакету треба освоїти досить складну систему позначень **Systems Dynamics**. Ця система реалізована у вигляді наступних конструкцій. Перша, названа рівнем, складається з таких компонентів, як гроші, складські запаси, шкідливі викиди і т.д. Друга конструкція – потік. Він поєднує елементи, які переміщуються між рівнями. Допоміжні атрибути і константи дозволяють модифікувати потік.

В процесі побудови моделі треба розміщати блоки і визначати перемінні, які становлять математичну основу кожного блоку. Для цього є діалогове вікно **Define Variable**, в якому утримується список всіх припустимих перемінних для кожного блоку і поле для опису перемінних, щоб користувачі легше могли зрозуміти цю модель.

Пакет **Powersim** виділяється здатністю обробляти масиви і підтримувати колективну роботу, а також тим, що містить бібліотеку з великим числом функцій. Масиви, що дозволяють використовувати набори перемінних, особливо зручні при побудові моделей з подібною структурою. Наприклад, при моделюванні роботи заводу, що має декілька ліній для виробництва кольорів різних квітів, можна змоделювати одну лінію і присвоїти їй масив з інформацією про кольори фарби. Це значно спростить побудову моделі і полегшить її розуміння. Масиви

зручні також для створення моделей, у конструкції яких рівні міняють свій стан, а розроблювач хоче простежити за цими змінами.

Пакет **Powersim** містить у собі понад 150 функцій, розділених на 16 груп, у тому числі фінансову, математичну, статистичну, графічну й історичну. Ключові параметри, діаграми і таблиці можна виводити безпосередньо на екран моделювання, спрощуючи тим самим перегляд результатів. Функція **Multiuser Game** надає можливість декільком користувачам одночасно запускати модель, щоб спільно над нею працювати. Це особливо корисно для проведення колективного тестування.

Powersim містить багато стандартних засобів **Windows- приложень**, такі як меню й інструментальні лінійки, і підтримує технології **Dynamic Data Exchange (DDE)** і **Object Linking and Embedding (OLE)**, що дозволяє вмонтувати **Powersim-Модель** в документ, створений текстовим процесором, так, щоб зміни в моделі автоматично відображалися в документі.

Пакет має багато прикладів моделей, що дозволяє новачкові швидко навчитися працювати з ним і використовувати всі його можливості.

13.2.3 Пакет **Ithink**

Пакет **Ithink** – один з найбільш потужних з розглянутих продуктів. По безперервному моделюванню він відстає від **Powersim**, однак краще підтримує дискретне моделювання. Крім того, пакет **Ithink** має навчальну програму, якісну документацію і велику кількість блоків для складання моделі. Крім базової версії, пакет має авторську версію, яка дозволяє включати в модель лінійки із движками та інші засоби керування моделлю, а також вводити діаграми та інші зображення прямо в модель, щоб користувачі могли контролювати процес моделювання і відразу бачити його результати.

Подібно **Powersim**, пакет **Ithink** використовує систему позначень **Systems Dynamics**, що в основному орієнтована на безперервне моделювання. Для реалізації цієї системи служать конструкції чотирьох типів: **станції**, аналогічні рівням у пакеті **Powersim**, **потоки**, **конвертери**, що нагадують допоміжні атрибути з **Powersim**, і **з'єднувачі**, що відповідають зв'язкам. Щоб створювати дискретні моделі, в пакеті є три спеціальні станції: **черги**, в яких елементи обробляються

за принципом «першим прийшов – першим обслужений», **сховища**, які перед початком обслуговування накопичують задану кількість елементів і зручні при пакетній обробці, **транспортери**, які передають елементи між станціями.

Моделі, побудовані за допомогою **Ithink**, складаються з рівнів і ієрархій. Користувач будує опис моделі на високому рівні за допомогою середовищ моделювання процесів, кожне з яких дозволяє створити модель однієї підсистеми, наприклад такий, як витрати ракетного палива в космічному кораблі. Завершивши опис, треба перейти на наступну ступінь деталізації і вводити в кожну підмодель необхідні конструкції. Між підмоделями встановлюються зв'язки, що вказують на їх взаємодію, приміром, на залежність між тим, наскільки правильно компанія склала рахунки для оплати, і кількістю телефонних дзвінків у відділ обслуговування покупців.

Побудувавши модель з необхідним числом ієрархічних рівнів, треба перейти до визначення математичних зв'язків між станціями, потоками та іншими конструкціями, для чого пакет пропонує список припустимих перемінних. Також він забезпечує проведення аналізу **чутливості** моделі шляхом її багаторазового запуску з різними вхідними параметрами.

Результати кожного прогону виводяться в окремому рядку вихідної діаграми. Це є основні види розподілів, застосовуваних для статистичного аналізу або діаграми. При виконанні моделі використовуються засоби анімації, що переміщують розташовані на різних рівнях станції відповідно до логіки моделі. Результати моделювання виводяться у вигляді часових діаграм або діаграм розкиду. Щоб задати діаграму або таблицю, потрібно вибрати використовувані **величини** й указати необхідні **параметри**.

Таблиці і діаграми можна переглядати в спеціальних вікнах пакета або у вікні моделі. Вибір форматів для виводу результату в **Ithink** менший, ніж **Extend**, але перевершує як **Powersim**, так і **Process Charter**. Пакет постачається з посібником з моделювання за назвою «Системне мислення» і спеціальною інформацією для користувачів, що працюють в різних галузях бізнесу.

Пакет **Ithink** підтримує меншу кількість функцій, ніж пакет Powersim.

13.2.4 Пакет **Extend**

Пакет **Extend** є найбільш потужним продуктом з розглянутих тут. Це - універсальний засіб моделювання, зручний при реорганізації бізнес-процесів. Для створення моделей у пакеті застосовується блокове середовище розробки, користуватися якою набагато простіше, ніж системою позначень **Systems Dynamics** для пакетів **Powersim** і **Ithink**, хоча трохи складніше, ніж блок-схемами продукту **Process Charter**. Пакет **Extend** має засоби побудови безперервних і дискретних моделей, широкий діапазон сформованих заздалегідь блоків, підтримує сторонніх постачальників і має можливість розширення. Він вперше з'явився на ринку в 1987 р. Спочатку він був орієнтований на інтерфейс комп'ютерів **Macintosh**, потім перенесений у середовище **Windows** за допомогою інтерфейсу прикладного програмування **Win32** і тепер може виконувати його інсталяцію. Пакет випускається в чотирьох версіях: **Basic**, **Extend+BPR** (Business Process Reengineering), **Extend+Manufacturing**, **Extend+BPR+Manufacturing**.

Додаткові засоби **BPR** і **Manufacturing** включають ряд важливих функцій для реінжинірингу підприємств. Крім того, існує багато продуктів сторонніх компаній, що підтримують **Extend** і орієнтованих на певні області застосування. До найбільш популярних з них належать програмні продукти для управляючих систем, а також для масового виробництва.

В базовий пакет **Basic** входить понад 90 сформованих блоків, об'єднаних у бібліотеки, з яких найчастіше використовуються **Discrete-Event**, **Generic** і **Plotter**. Бібліотека **Discrete-Event** містить у собі різні дії, черги, шлюзи й таймери. Бібліотека **Generic** містить генератори випадкових чисел і вихідні дані, файли для вхідної і вихідної інформації, а також блоки для математичних, булевих і фінансових даних. Бібліотека **Plotter** складається з блоків для створення вихідних діаграм і таблиць. Інші бібліотеки мають спеціальне призначення, наприклад, здійснюють збір статистичної інформації.

Пакети **BPR** і **Manufacturing** мають додаткові бібліотеки. Крім того, в **Extend** є вбудована мова **Modl**, що дозволяє будувати спеціалізовані блоки. Вибором блоку з бібліотеки **Discrete-Event** автоматично будується дискретна

модель; у протилежному випадку – **безперервна** модель. Блоки можуть забезпечуватися вхідними і вихідними з'єднувачами. При побудові зв'язків не потрібно вдаватися до спеціальних засобів об'єднання блоків, прийнятих в інших продуктах. Користувач за допомогою миші просто малює на екрані лінії, що зв'язують вихідний з'єднувач одного блоку з вхідним з'єднувачем іншого. Якщо зв'язок неприпустимий, **Extend** не приймає його. Для перегляду вихідної інформації, що надходить з блоку треба поєднати один з його вихідних з'єднувачів із блоком з бібліотеки **Plotter**. Такі блоки мають високий ступінь гнучкості й можуть приймати безліч типів вхідних даних, виводячи їх у вигляді таблиць або діаграм.

При побудові моделі центра обслуговування за допомогою **Extend** можна практично необмежено міняти частоту надходження замовлень, час обслуговування одного клієнта та інші параметри. Модель будується швидко і автоматично виводить результати. Пакет дає можливість проводити аналіз чутливості за декількома перемінними, що беруть участь у моделюванні, і багаторазово виконувати модель для різних значень перемінних. Результати кожного прогону виводяться на графобудівник у різних кольорних режимах. Це дозволяє розроблювачеві досліджувати модель при різних обставинах і швидко прийняти рішення.

Пакет **Extend** має функції створення **авторських моделей**, за допомогою яких розроблювач включає у вікно моделі текст, геометричні зображення й управляючі блоки, щоб користувачі могли самостійно модифікувати модель. Для контролю за процесом моделювання і виводу результатів на дисплей застосовуються засоби, організовані за принципом блокнота, що зручніше у використанні, ніж електронна таблиця **Key Values** пакета **Process Charter**.

До пакету надаються детальні вказівки користувачеві, навчальна програма і приклади моделей з різних сфер діяльності, які можуть бути основою для створення нових моделей, що полегшує процес моделювання.

Слід зауважити, що пакет **Extend** використовується в повному обсязі тільки на комп'ютерах типу **Macintosh** і має високу вартість.

13.2.5 Пакет **Pilgrim**

Пакет **Pilgrim** підтримує дискретно-безперервне моделювання [10]. Він був створений у Росії в Московському державному університеті економіки, статистики й інформатики (МЕСІ) у співробітництві з декількома комп'ютерними фірмами.

В концепції пакета використовуються поняття граф моделі, транзакт, вузли графа, події, ресурс. **Граф моделі** – це спрямований граф, що поєднує всі процеси. **Транзакт** – це формальний запит на яке-небудь обслуговування. Він має набір властивостей і параметрів, що динамічно змінюються. Шляхи міграції транзактів по графу стохастичної сіті визначаються логікою функціонування компонентів моделі у вузлах графу сіті, які є центрами обслуговування транзактів. В цих центрах транзакти можуть обслуговуватися й затримуватися, а вузли можуть породжувати сімейства нових транзактів і знищувати інші транзакти. Вихід з вузла одного транзакту називається **подією**. Вони завжди відбуваються в певні моменти часу і можуть бути зв'язані із місцем простору. **Ресурс** може характеризуватися потужністю, залишком і дефіцитом, незалежно від його природи в процесі моделювання. Матеріальні ресурси підрозділяються на непереміщувані і переміщувані. Непереміщуваний ресурс пов'язаний з певним місцем. Це може бути, наприклад, конкретний службовець із обслуговуючого персоналу, який не може переміщатися разом із клієнтом (транзактом). Після обслуговування одного клієнта він або буде обслуговувати наступного, якщо є черга, або буде простоювати. Вузли, транзакти і ресурси можуть бути прив'язані до місць **простору** і мігрувати в ньому.

В пакеті **Pilgrim** є 17 типів вузлів, які дають можливість моделювання просторової динаміки, роботи з ресурсами (грошима і матеріальними цінностями, рахунками бухгалтерського обліку, банківськими рахунками). В його **конструкторі моделей** використовується CASE-технологія багат шарового імітаційного моделювання. Модель у цьому конструкторі можна представити як набір наступних компонентів: граф моделі, параметри ініціалізації моделі, перемінні моделі, фрагменти програмного коду мовою C++, включені в модель. Конструктор

моделей пакету **Pilgrim** забезпечує можливість проведення структурного системного аналізу, результатом якого є багаторівнева ієрархічна декомпозиція глобального процесу з розкладом на компоненти і представленням кожного рівня у вигляді графічного шару. Крім того, він дає можливість колективного управління процесом моделювання і створення користувальницьких блоків мовою програмування C++. При цьому він має інтерфейси з базами даних і автоматично генерує програмний текст моделі. Слід відмітити, що при такій розвинутій функціональності цей пакет має невисоку вартість. До його недоліків відносять складну систему позначень і необхідність знати мову програмування C++.

13.2.6 Пакет Vensim

Пакет **Vensim** призначений для побудови системно-динамічних моделей, аналогічних пакетам **Powersim** і **Ithink**. Він підтримує безперервне моделювання, має засоби оптимізації і статистики; можливість створювати DLL-бібліотеки, які підключаються до інших програм. Має простий графічний інтерфейс, розрахований на професіоналів, і розширювану бібліотеку функцій. Недоліком пакета **Vensim** є відсутність можливості конвертування даних і невелика кількість вбудованих математичних функцій у версії PLE.

ЧАСТИНА 6

ДОСВІД ІМІТАЦІЙНОГО МОДЕЛЮВАННЯ БІЗНЕС-СИСТЕМ

Розділ 14

Приклади імітаційного моделювання бізнес-систем

14.1 Моделювання впливу зміни процентної ставки на сальдо банківського рахунку

Цей приклад, взятий з [48], демонструє найпростішу модель річного росту сальдо банківського рахунку, швидкість якого залежить від коливної щомісячної процентної ставки. Відомо, що після чергового інтервалу нараховані за інтервал відсотки не виплачуються, а приєднуються до грошової суми, наявної на початок цього інтервалу. Для визначення нарощеної суми застосовують формули складних відсотків. Такі складні позичкові відсотки сьогодні є досить розповсюдженим видом процентних ставок.

Для початку була створена схема моделі, показана на **рис.14.1**, де розміщені **резервуар**, **потік** і **конвертер**, і їм присвоєні відповідні назви. Крім того, розміщені об'єкти **графік** і **таблиця**, в яких будуть відображатися результати роботи моделі.

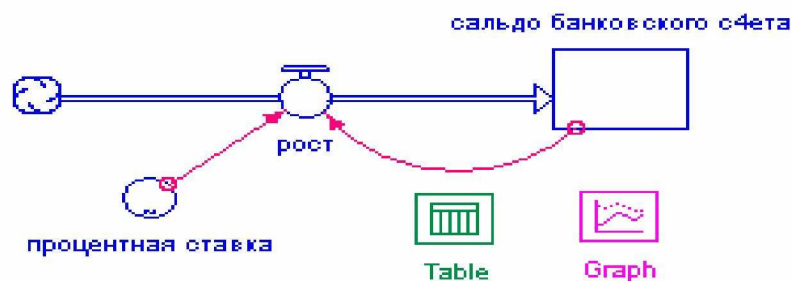


Рис.14.1 Схема моделі

Для переходу на етап чисельного опису треба натиснути на кнопку із зображенням **земної кулі**. На кожному елементі схеми з'явиться зображення

знака питання. Подвійне клацання по конвертеру відкриває діалогове вікно визначення числових параметрів конвертера. В полі ініціалізації вказується, що буде використовуватися функція **TIME** (вона вибирається зі списку **Builtins**). Натиск на кнопку **Become Graphical Function** задає процентну ставку залежно від модельного часу, у цьому випадку місяця. З'являється діалогове вікно визначення графічної функції (рис.14.2), в якому треба вказати значення процентної ставки і валюту Euro.

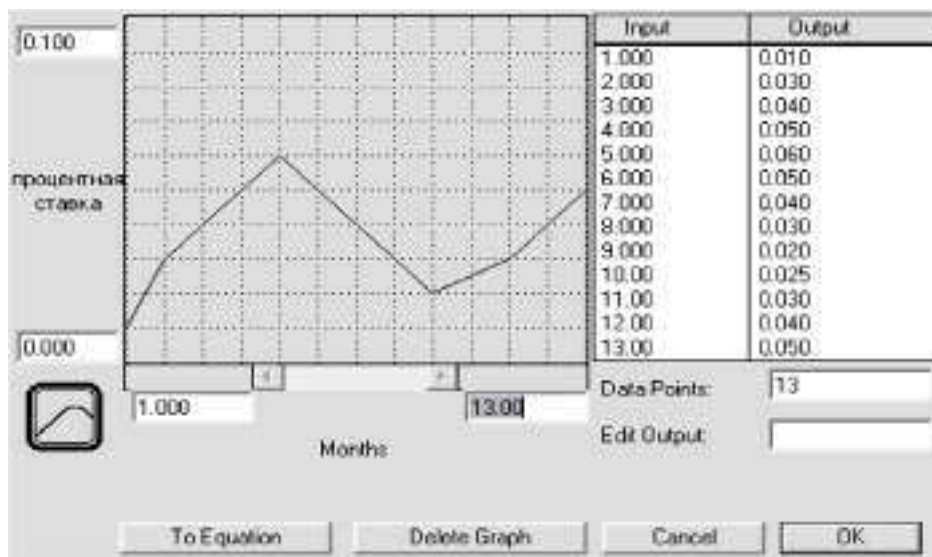


Рис.14.2 Визначення графічної функції

Після подвійного клацання по резервуару з'явиться вікно рис.14.3.

Рис.14.3 Введення початкового значення сальдового рахунку

В ньому було задано початкове значення сальдового рахунку, рівне 10000, і одиниця виміру валюти (нажати кнопку Units і у вікні, що відкривається, зі списку, що випадає, вказати Euro).

Після цього треба відкрити діалогове вікно визначення числових параметрів потоку для задання формули росту сальдо за допомогою перемінних з списку **Required Inputs**, й калькулятора (рис.14.4).

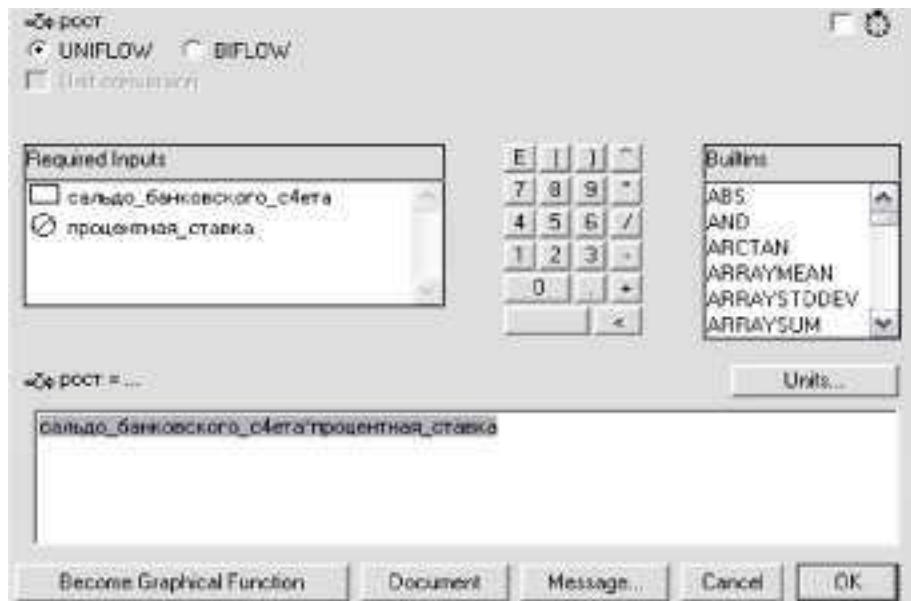


Рис.14.4 Завдання формули росту сальдо

Для визначення **графіка** викликається його вікно. Після подвійного клацання по ньому з'являється діалогове вікно визначення графіка. У ньому вказується назва графіка і вибираються перемінні, які треба нарисувати на графіку (рис. 14.5) за допомогою кнопок. Для визначення таблиці треба відкрити діалогове вікно (рис.14.6) і в ньому вибрати ті перемінні, значення яких треба ілюструвати таблицею, і задати інші параметри таблиці. Автоматично буде згенеровано програмний код. Для переходу на його рівень, щоб побачити текст програми, треба клацнути на стрілці долілиць ліворуч від робочого аркуша. Для її запуску треба вибрати з меню пункт Run. Результат роботи моделі буде відбито у таблиці і на графіку (рис.14.7).

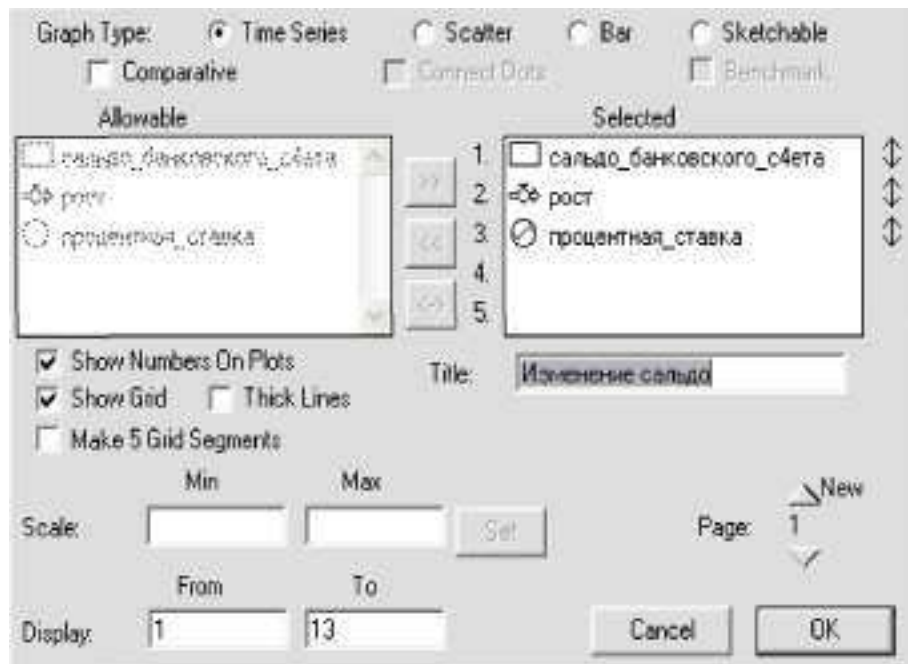


Рис.14.5 Визначення графіка зміни сальдо

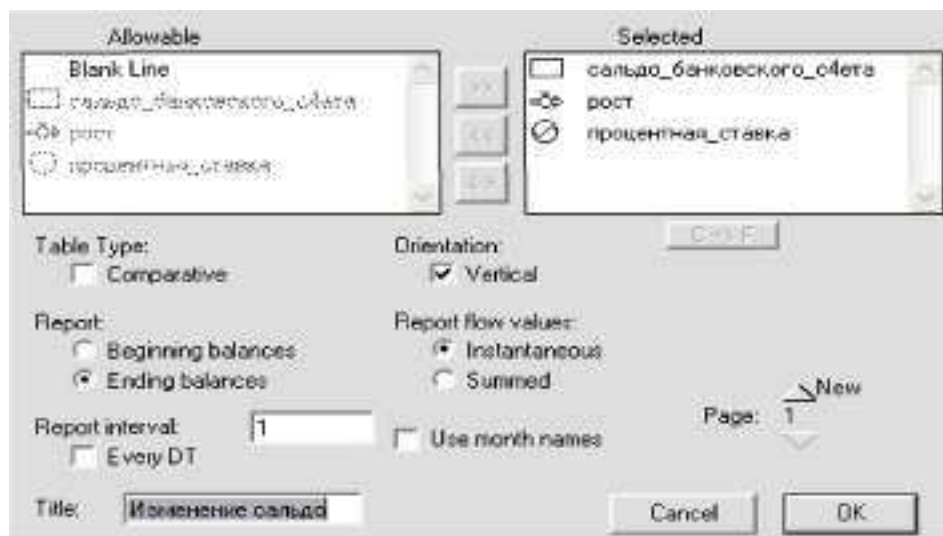


Рис.14.6 Визначення таблиці

□ $\text{сальдо_банковского_счета}(t) = \text{сальдо_банковского_счета}(t - dt) + (\text{рост}) * dt$

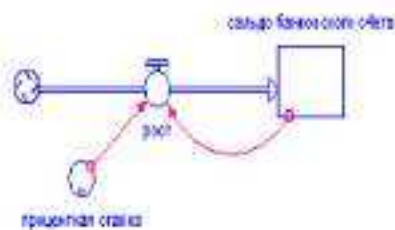
INIT сальдо_банковского_счета = 10000

INFLOWS:

⊗ $\text{рост} = \text{сальдо_банковского_счета} * \text{процентная_ставка}$

⊗ $\text{процентная_ставка} = \text{GRAPH}(\text{TIME})$

(1.00, 0.01), (2.00, 0.03), (3.00, 0.04), (4.00, 0.05), (5.00, 0.06), (6.00, 0.05), (7.00, 0.04), (8.00, 0.03),
(9.00, 0.02), (10.0, 0.025), (11.0, 0.03), (12.0, 0.04), (13.0, 0.05)



Модель учитывает изменение процентной ставки на сальдо банковского счета

Месяц	сальдо банковского счета	рост	процентная ставка
январь	10000,00		0,01
1	10179,11	382,82	0,03
2	10623,62	390,88	0,04
3	10961,64	515,99	0,04
4	11594,78	667,25	0,05
5	12280,64	626,28	0,06
6	12807,60	629,82	0,05
7	13169,62	429,82	0,04
8	13562,34	395,68	0,03
9	13983,68	329,68	0,02
10	14042,73	489,41	0,025
11	14823,64	651,82	0,03
12	15483,68	737,28	0,04

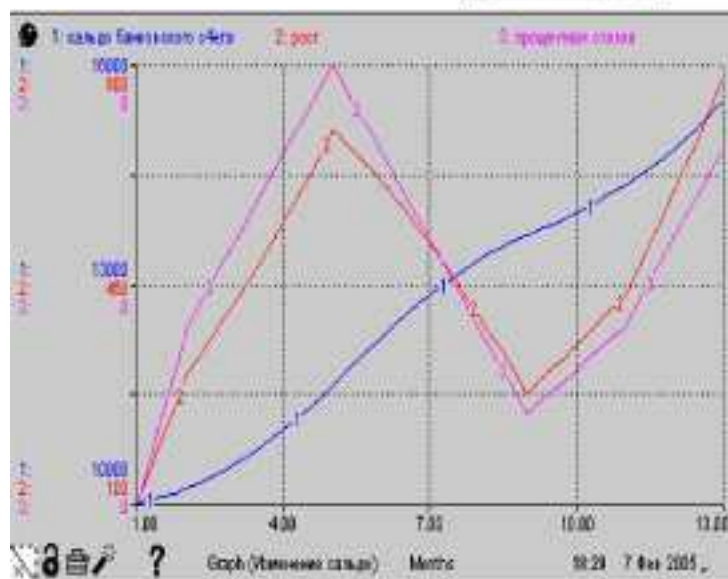


Рис.14.7 Результаты работы модели

14.2 Моделювання розподілу доходів фірми від продажу і від сервісного обслуговування

Необхідно визначити доцільність і перспективність орієнтації фірм, що провадять комп'ютерне устаткування, на наступну концентрацію своїх зусиль у сфері надання сервісних послуг із супроводу. На **рис.14.8** показана схема моделі для вирішення цього завдання, взята з [48]. Ця модель універсальна і може використовуватися в найрізноманітніших областях застосування поточкових схем. Вона виникла після дослідження процесів, що відбуваються в комп'ютерному бізнесі після буму 60-х, коли в США відбулося насичення ринку обчислювальним устаткуванням, і коли в умовах найжорстокішої конкуренції деякі фірми, що динамічно розвиваються, звернули увагу на новий привабливий сектор ринку. Далі модель в уточненому вигляді знайшла своє застосування в інших сферах ділової активності і стала класичною.

Потік hardware sales ескізно відображає динаміку бізнес-процесу продажу наукомісткого обчислювального устаткування і відповідного базового системного й прикладного програмного забезпечення, якому передують цілий комплекс заходів і досліджень маркетингового характеру.

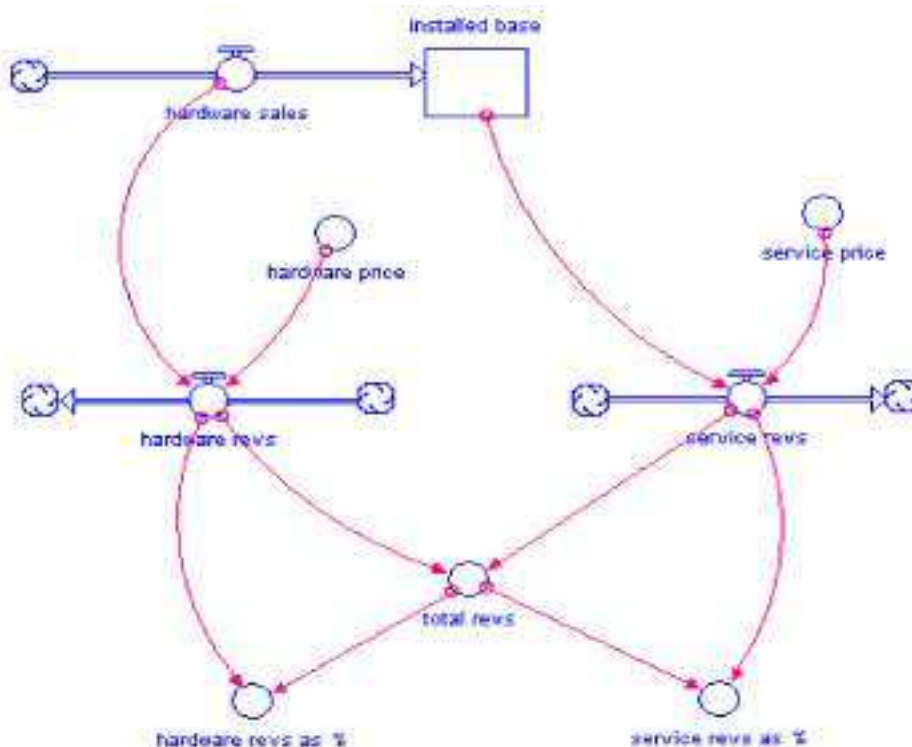


Рис.14.8 Схема моделі бізнес-процесів

Фонд installed base припускає наявність парку, що розширюється, комп'ютерів, які потенційно бідують у постійному сервісному обслуговуванні. На відеограмі видно, що потік hardware revs залежить від потоку hardware sales і конвертора hardware price, а потік service revs - від потоку service price і вмісту фонду installed base.

Інтенсивності цих двох симетричних потоків управляються шляхом обчислення значень відповідних виражень, які треба задавати при визначенні параметрів схеми. Ця модель не враховує можливі коливання реальних цін і необхідно буде обмежитися лише модельними припущеннями.

Всі доходи фірми збираються в конвертері total revs і потім пропорційно розділяються на hardware revs і service revs в %.

Метою імітаційних експериментів на моделі є визначення динаміки можливих співвідношень між вихідними значеннями відповідних конвертерів залежно від коливання цін на комп'ютери, зміни вартості сервісу, а також збільшення парку встановленого устаткування. Уточнення схеми проводиться на фазі остаточного формування поведінкової моделі. Вихід на цей режим керування здійснюється через кнопку на лівій стороні діалогового вікна.

Далі подвійним клацанням миші викликаються відповідні діалогові вікна і в них визначаються значення констант. Потім або формуються алгебраїчні вираження, або задаються можливі початкові значення, або задаються співвідношення через графічні функції. В цьому випадку задається значення потоку hardware sales. Діалогове вікно визначення графічної функції набуває вигляд **рис.14.9**.

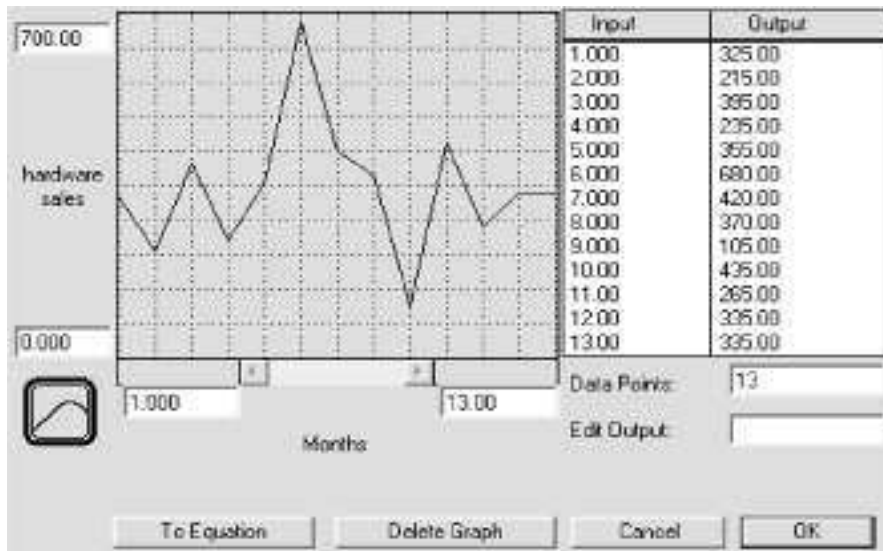


Рис.14.9 Визначення графічної функції

Аналогічно визначаються числові параметри конверторів hardware price і service price.

Визначення початкового значення резервуара installed base показано на **рис. 14.10**, (в цьому випадку це кількість проданих комп'ютерів),.

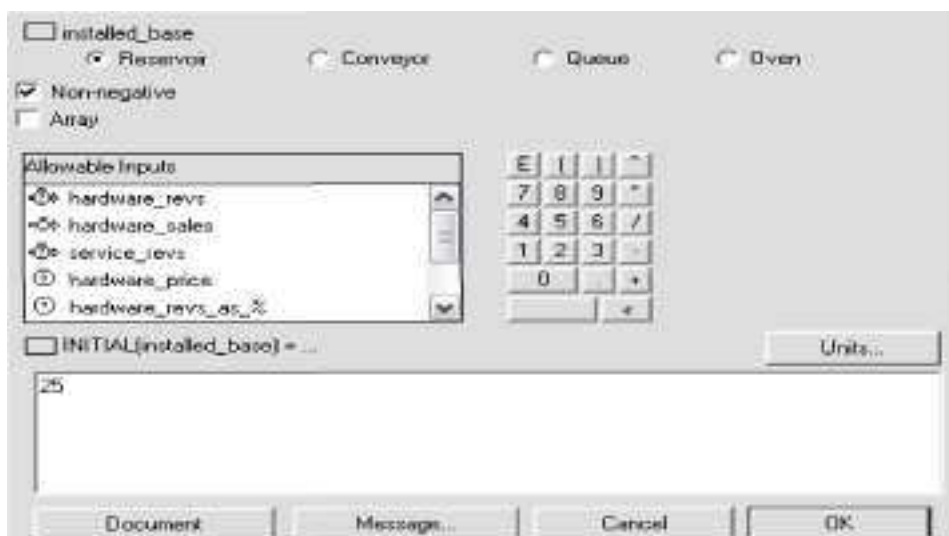


Рис.14.10 Визначення початкового значення

Визначення значень потоків hardware revs і service revs здійснюється через діалогове вікно визначення числових параметрів потоку. В поле ініціалізації вибираються блоки з списку **Required Inputs** і калькулятору задається формула. Аналогічно задається також потік **service revs**.

Діалогове вікно визначення числових параметрів конвертора **total revs** показано на **рис.14.11**.

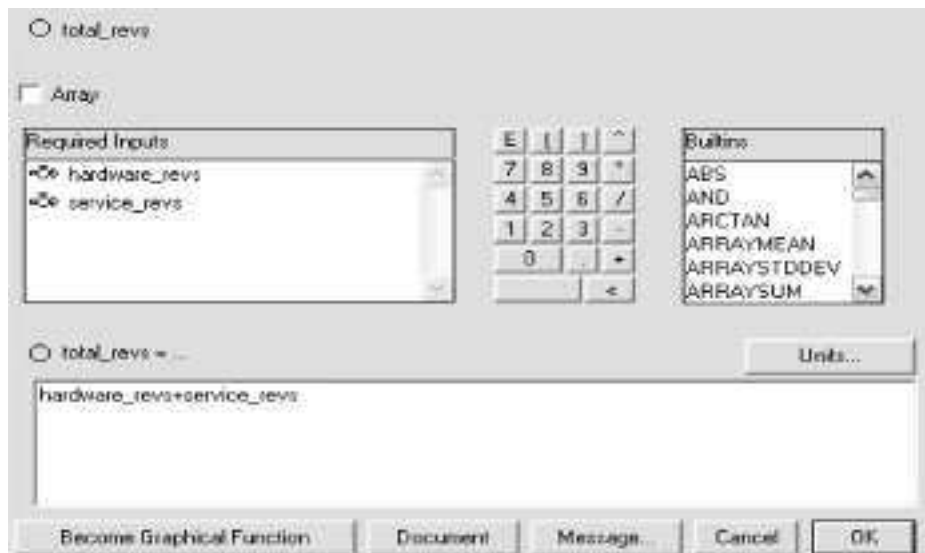


Рис.14.11 Діалогове вікно визначення числових параметрів конвертора

Діалогові вікна для визначення конверторів **service revs as %** і **hardware revs as %** мають аналогічний вигляд.

Про закінчення всіх можливих визначень елементів моделі можна переконатися по відсутності зображень фондів, потоків і конверторів з попереджувачими знаками «?»). Для того щоб побачити результати, треба в модель додати об'єкт **графік**. Його властивості визначаються за допомогою діалогового вікна, зображеного на **рис.14.12**.

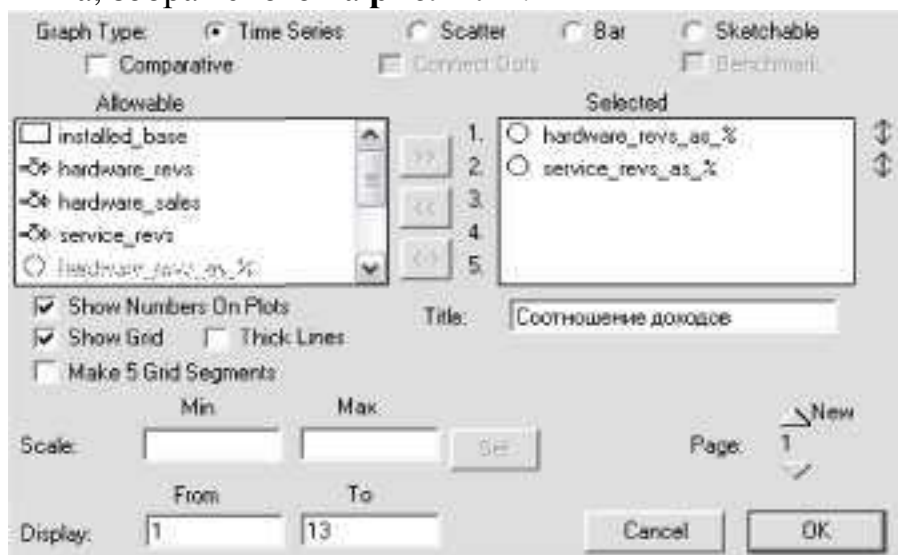


Рис.14.12 Визначення властивостей графіка

Далі треба додати в модель **таблицю**, в якій будуть відображатися зміни всіх елементів моделі. Діалогове вікно визначення об'єкта таблиця показана на **рис.14.13**.

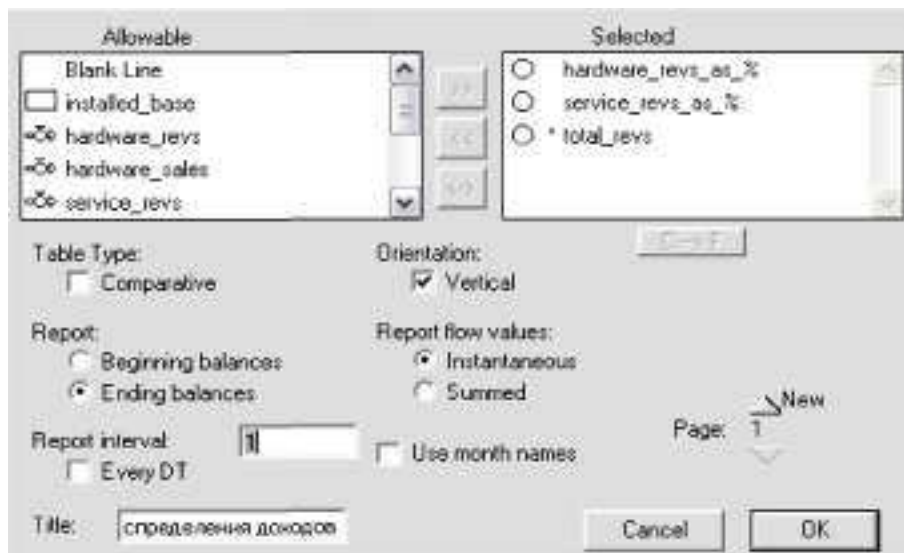


Рис.14.13 Визначення об'єкта таблиця

Встановивши необхідні для проведення імітаційних експериментів налаштування або використавши готові за замовчуванням, можна через меню **Run** запустити режим імітації. Відслідковуючи просування процесу в інтервалі модельного часу, що задається, через індикатор, треба почекати одержання результату імітаційного експерименту. Результат, що демонструє загальну картину поведінки модельних одиниць, наведено на **рис.14.14**. Таблиця буде виглядати, як показано на **рис.14.15**.

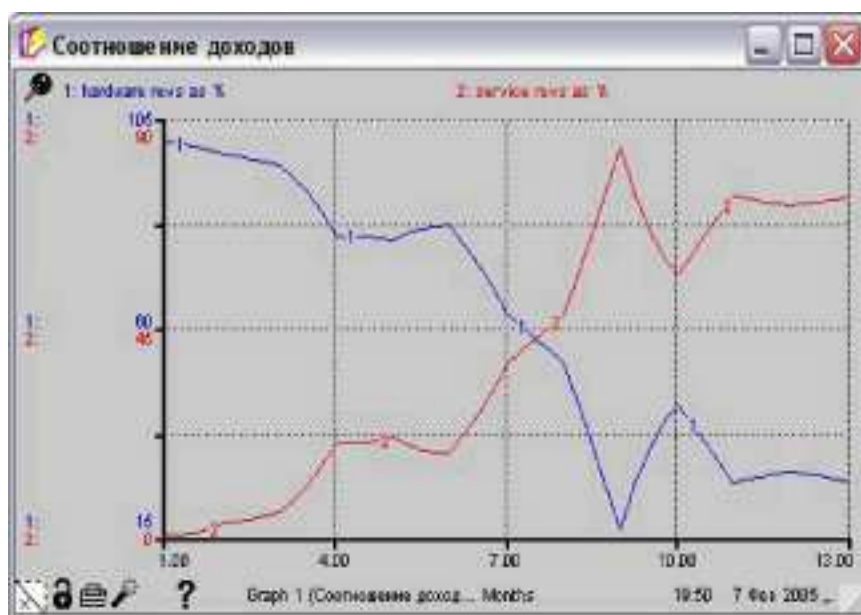


Рис.14.14 Результати імітаційного експерименту

Months	hardware revs as	service revs as %	total revs
1	97,40	2,60	1134043,20
2	94,88	5,11	1446286,00
3	79,66	20,34	1188859,38
4	78,70	21,30	1270468,76
5	81,84	18,06	2172389,72
6	69,13	36,87	1988445,31
7	62,41	47,59	1774853,13
8	16,41	83,59	1500022,66
9	43,53	56,47	2082324,22
10	26,53	73,47	2088630,06
11	28,01	71,08	2218828,81
12	26,96	73,04	2289343,76

Рис.14.15 Модель розподілу доходів

Схеми можна декомпонувати на під схеми і інкапсулювати їх у відповідні фрейми секторів, як показано на **рис.14.16**.

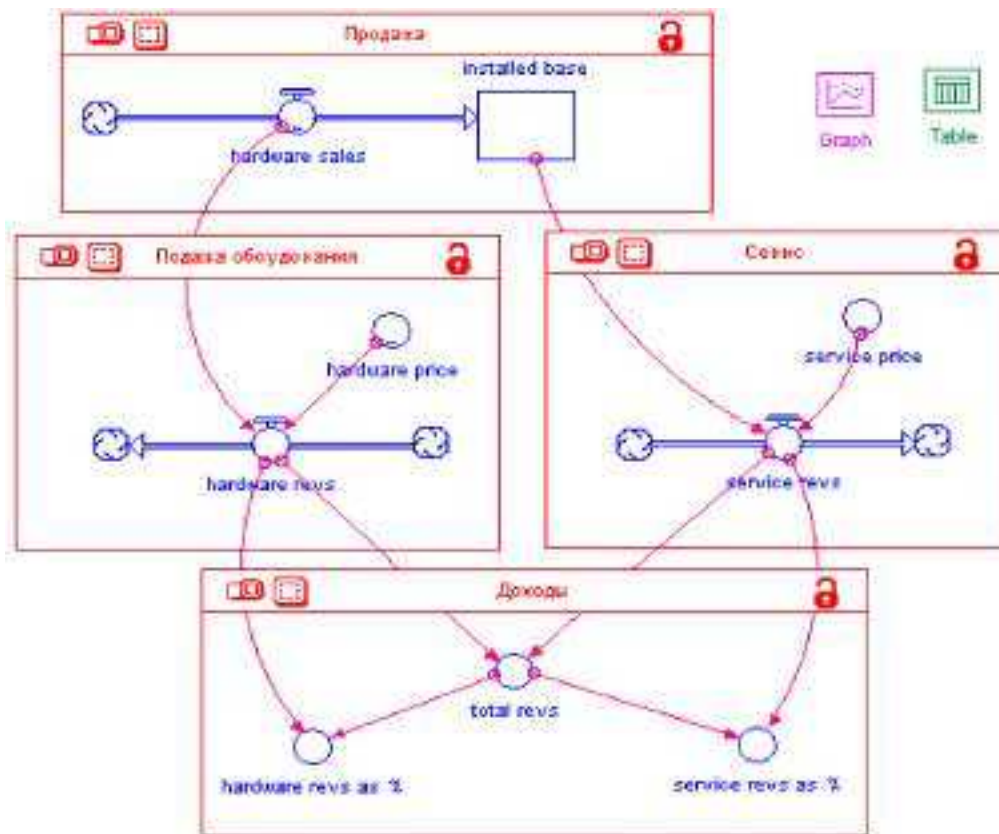


Рис.14.16 Декомпозиція схеми моделі

Якщо після цього піднятися за допомогою стрілки на правій панелі керуючого вікна нагору, то на самому верхньому рівні подання моделі Ithink може запропонувати відображення фреймів процесів у вигляді традиційних блок-схем (**рис.14.17**).

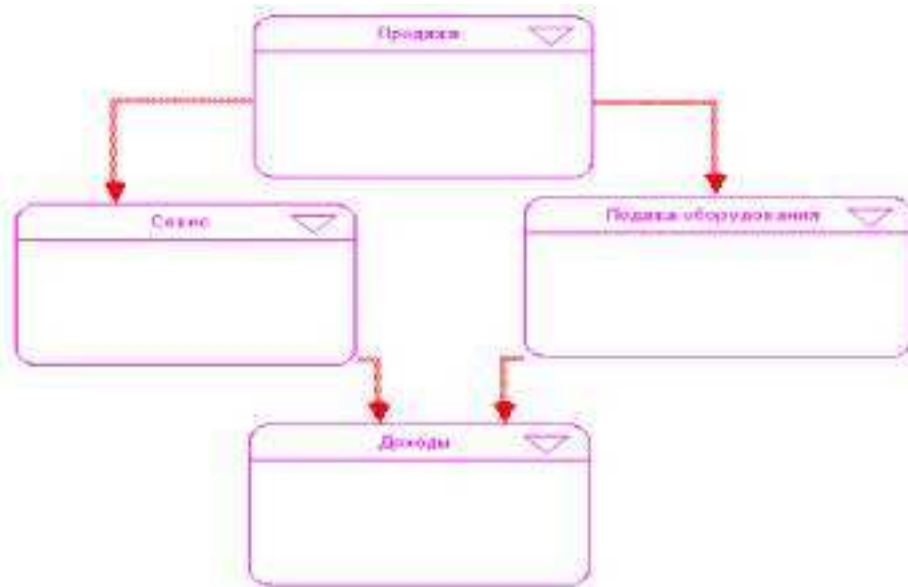


Рис.14.17 Традиційні блок-схеми

Спустившись за допомогою стрілки на правій панелі керуючого вікна, можна на самому нижньому рівні подання моделі у вікні побачити приведений нижче програмний код моделі. Для цього в пункті меню Interface Prefs на рівні інтерфейсу повинен бути поставлений прапорець Link High-Level Map to Model.

Доходи

- hardware_revs_as_% = 100*hardware_revs/total_revs
- service_revs_as_% = 100*service_revs/total_revs
- total_revs = hardware_revs+service_revs

Податки обладнання

UNATTACHED:

- hardware_revs = hardware_sales*hardware_price
- hardware_price = GRAPH(TIME)
(1.00, 4700), (2.00, 4575), (3.00, 3750), (4.00, 3550), (5.00, 2950), (6.00, 2975), (7.00, 2725), (8.00, 2500), (9.00, 2225), (10.0, 2300), (11.0, 2025), (12.0, 1975), (13.0, 1950)

Продажа

installed_base(t) = installed_base(t - dt) + (hardware_sales) * dt

INIT installed_base = 25

INFLOWS:

- hardware_sales = GRAPH(TIME)
(1.00, 325), (2.00, 215), (3.00, 395), (4.00, 235), (5.00, 355), (6.00, 680), (7.00, 420), (8.00, 370), (9.00, 105), (10.0, 435), (11.0, 265), (12.0, 335), (13.0, 335)

Сервіс

UNATTACHED:

- service_revs = installed_base*service_price
- service_price = GRAPH(TIME)
(1.00, 19.5), (2.00, 85.0), (3.00, 135), (4.00, 230), (5.00, 235), (6.00, 265), (7.00, 295), (8.00, 315), (9.00, 405), (10.0, 410), (11.0, 420), (12.0, 425), (13.0, 425)

Not in a sector

14.3 Моделювання кредитування підприємств

При плануванні роботи банку з групою пріоритетних клієнтів виникає необхідність розподілу між ними кредитних ресурсів. При цьому керівництво повинно визначити параметри кредитування - строки, графік, ставку відсотка та інші умови. Для вирішення таких завдань, що вимагають перебору великої кількості різних варіантів, використовуються спеціалізовані банківські програмні засоби з твердим інтерфейсом. Звичайно такі програми є досить дорогими і вузько націвленими.

На відміну від них система **Ithink** має більшу гнучкість, і з її допомогою можуть бути вирішені як такі, так і зовсім інші завдання. Насамперед, необхідно зазначити, що це типово «потокowe» завдання. Потік коштів надходить із банку на підприємство, утримується протягом деякого часу, а потім або виводиться з моделі, або вертається в банк. Основною проблемою є розробка певної структури відносин між клієнтами – черговість, пріоритетність і визначення основних умов. При зміні цієї структури клієнти одержать уже інші кредити в інші строки. Загальний розмір виділених кредитних ресурсів також позначається на умовах кредитування окремих клієнтів.

Пакет **Ithink** дозволяє операторові змінювати пріоритети й умови кредитування, в результаті чого змінюються також графіки кредитування підприємств, і динаміка стану розрахункового рахунку банку. Все це дозволить аналітикові швидко переглянути різні варіанти кредитних схем і вибрати серед них оптимальну.

В даному прикладі для простоти в модель закладено наступні пріоритети [7]. Підприємство №1 – головний партнер банку. Воно одержує кредити в першу чергу. Якщо кредитні можливості банку перевищують потреби підприємства (10 млн. грн), то ресурси направляються підприємству №2. Максимальний розмір кредиту, що може одержати підприємство №2, також обмежений (10 млн. грн). Надлишок направляється підприємству №3.

Межа за розміром кредиту для третього підприємства становить 5 млн. грн. Серія кредитів представлена чотирма кредитами по 15 млн. грн, що надходять щокварталу. Першому підприємству кредит може видаватися на 3 квартали, другому – тільки на 2 квартали, третьому – на 1 квартал.

В даному завданні мова йде про деяку послідовність – потік кредитів. В пакеті **Ithink** існує елемент **конвеєр**, що є повним аналогом підприємства, яке приймає кредит. Грошова сума надходить до нього, утримується протягом певного строку, а потім вертається в загальний потік, тобто тут передбачається, що кошти, «минулі» через підприємства, виводяться із процесу.

Елемент **черга** моделює серію кредитів, тобто виконує в даній моделі функції банку. В ньому встановлюється пріоритетність розподілу кредитних ресурсів. У вигляді **потоків** кошти надходять на підприємства. Модель процесу кредитування на базовому рівні подання поточковими схемами, створена за допомогою структурних елементів пакета **Ithink**, наведена на **рис.14.18**.

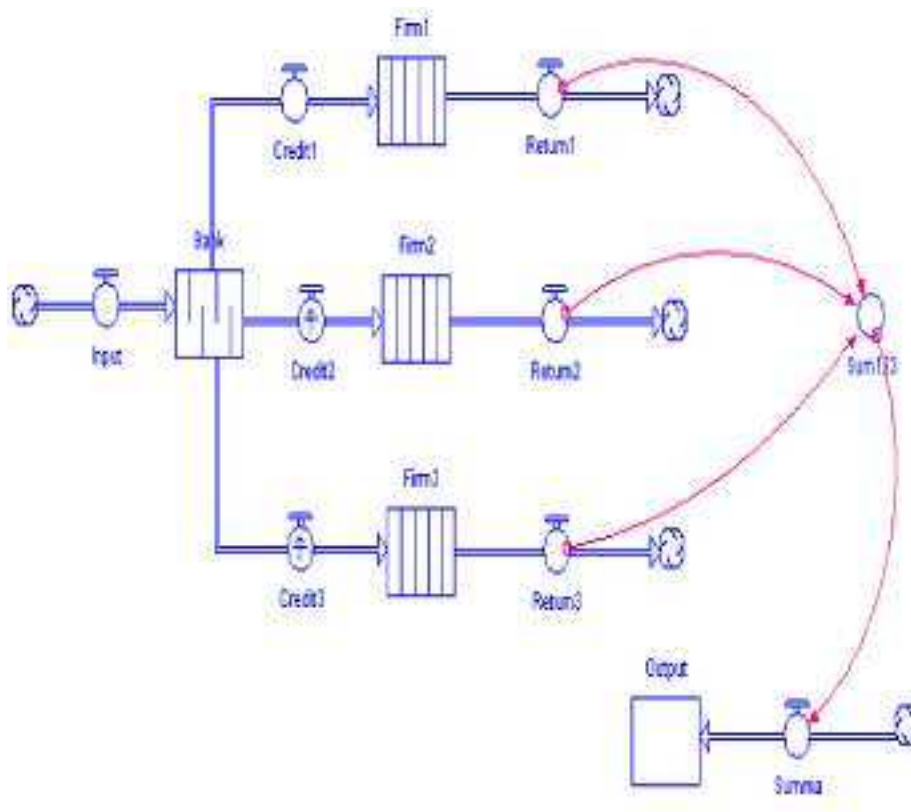


Рис.14.18 Модель процесу кредитування

Для введення в модель числових даних треба у вікні **Ithink** натиснути кнопку із зображенням **маркера у вигляді глобуса**. На моделі появляться **знаки питання** на тих блоках, параметри яких ми повинні визначити.

Для задання числових характеристик необхідне подвійне клацання лівою кнопкою миші по обумовленому об'єкту.

Завдання параметрів вхідного потоку **INPUT** здійснюється у вікні, показаному на **рис.14.19**.

Завдання параметрів черги здійснюється за допомогою вікна, для виклику якого треба клацнути по елементу **Bank**.



Рис.14.19 Завдання параметрів черги

Для визначення пріоритетів вихідних потоків треба в діалоговому вікні вихідних потоків із черги **Credit2** і **Credit3** поставити прапорець **Overflow**, після чого на зображенні потоку з'явиться **хрестик**.

Далі варто визначити параметри для кожного підприємства. Вікно для визначення структурного елемента **конвеєр Firm1** представлено на **рис.14.20**. Тут вказується час транзиту, обмеження на вхідний потік і ємність конвеєра, а також початкове значення блоку. Аналогічно визначаються також конвеєри **Firm2** і **Firm3**.

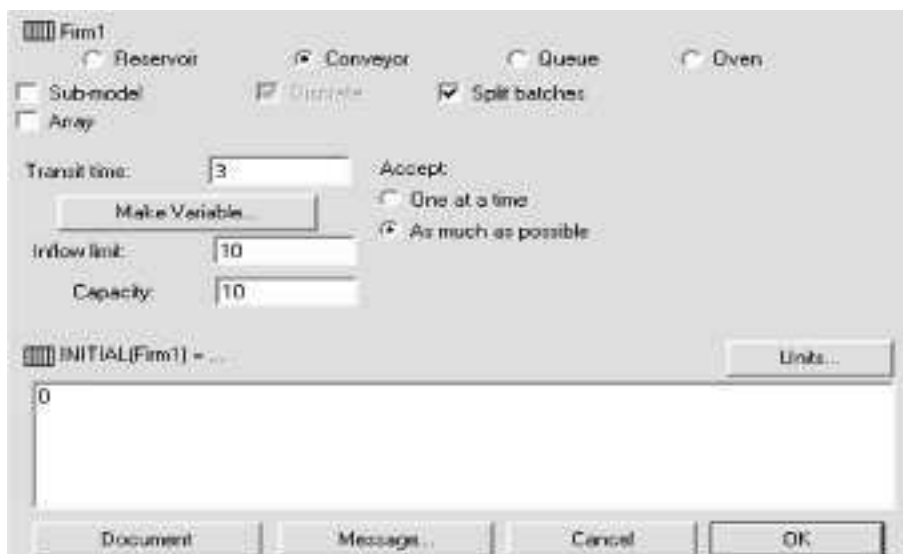


Рис.14.20 Визначення структурного елемента **конвеєр Firm1**

Після цього за допомогою діалогового вікна на **рис.14.21** можна перейти до ініціалізації елемента **SUM_123**. Для того, щоб підсумувалися значення потоків (за допомогою елемента **конвертер**) треба з списку **Required Inputs** вибрати і скласти необхідні елементи.

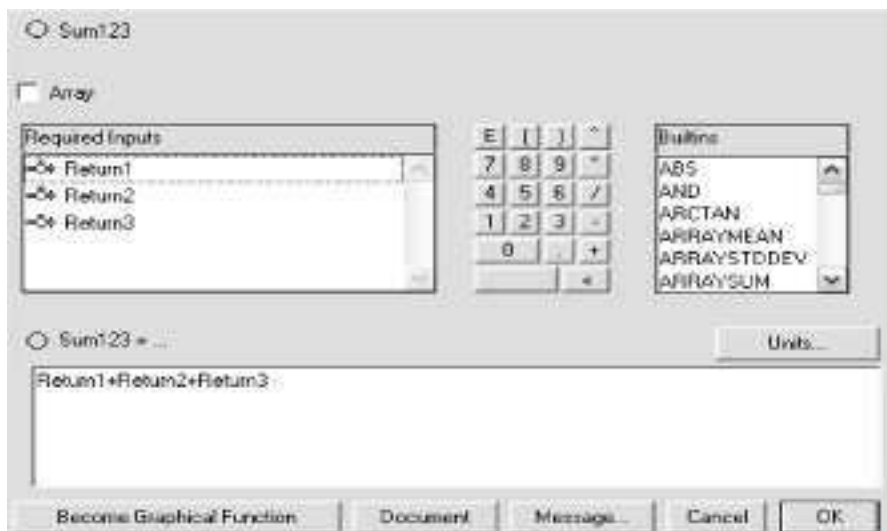


Рис.14.21 Ініціалізація елемента **SUM_123**

Аналогічно визначаються елементи моделі, що залишилися. В результаті всіх виконаних дій сформувався наступний програмний код:

```

EЯ BANK(t) = BANK(t - dt) + (INPUT - Credit_1 - Credit_2 - Credit_3) * dt
INIT BANK = 15,15,15,15
INFLOWS:
:*> INPUT = 0
OUTFLOWS:
:*> Credit_1 = QUEUE OUTFLOW
:*> Credit_2 = QUEUE OUTFLOW
:*> Credit_3 = QUEUE OUTFLOW
II FIRM_1(t) = FIRM_1(t - dt) + (Credit_1 - Return_1) * dt INIT FIRM_1 = 0
TRANSIT TIME = 3 INFLOW LIMIT = 10 CAPACITY = 10 INFLOWS:
:*> Credit_1 = QUEUE OUTFLOW OUTFLOWS: :*> Return_1 = CONVEYOR OUTFLOW
m FIRM_2(t) = FIRM_2(t - dt) + (Credit_2 - Return_2) * dt INIT FIRM_2 = 0
TRANSIT TIME = 2 INFLOW LIMIT = 5 CAPACITY = 10 INFLOWS:
:*> Credit_2 = QUEUE OUTFLOW OUTFLOWS: :*> Return_2 = CONVEYOR OUTFLOW
III FIRM_3(t) = FIRM_3(t - dt) + (Credit_3 - Return_3) * dt INIT FIRM_3 = 0
TRANSIT TIME = 1 INFLOW LIMIT = 5 CAPACITY = 5 INFLOWS:
:*> Credit_3 = QUEUE OUTFLOW OUTFLOWS: :*> Return_3 = CONVEYOR OUTFLOW
n OUTPUT(t) = OUTPUT(t - dt) + (Sum123) * dt INIT OUTPUT = Sum123 INFLOWS:
■:*> Sum123 = SUM_123 ° SUM 123 = Return 1+Return 2+Return 3

```

Щоб побачити результати роботи моделі, треба на базовому рівні помістити **графіки** і зв'язати їх з відповідними елементами моделі, а потім вибрати пункт меню **Run**. На виході моделі буде зазначений, зокрема, розмір

кредитних ресурсів, отриманих кожним підприємством. При заданих пріоритетах кредити розподілилися по підприємствах таким чином:

№1 - 20 млн. грн, №2 - 25 млн. грн, №3 - 15 млн. грн.

Графік повернення коштів банку наведено на **рис.14.22**.



Рис.14.22 Графік повернення коштів банку

При аналізі результатів необхідно враховувати, що потік кредитів визначається не тільки їхнім обсягом, але й строком кредитування.

Кредитні потоки для підприємств показано на **рис.14.23-14.25**. Тут по горизонтальній осі відкладено час, по вертикальній осі – кількість кредитних ресурсів, які перебувають у розпорядженні підприємства на даний час.

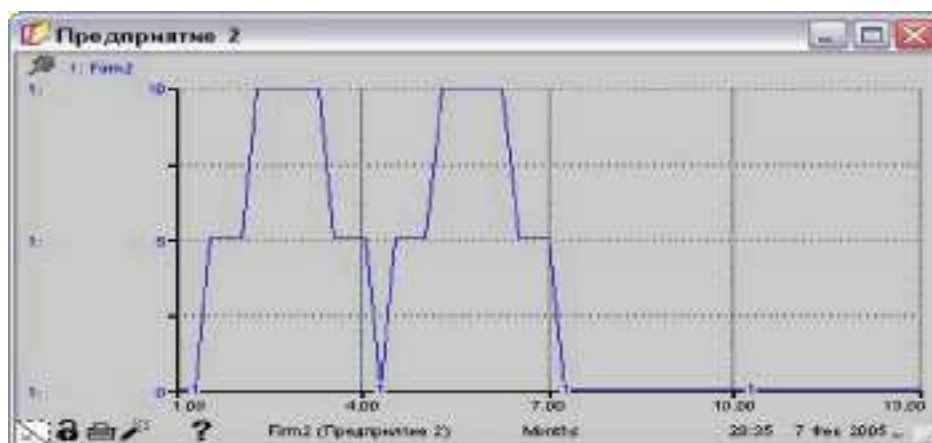


Рис.14.23 Кредитний потік для підприємства №2

Положення підприємства №2, з погляду строків і кількості отриманих ресурсів, явно переважніше, ніж у підприємства №3. Сума отриманих ним

ресурсів доходила до 25 млн.грн, хоча й надходила меншими порціями, а у підприємства №3 вона склала, без обліку терміновості, 15 млн. грн, тобто вона навіть більша, ніж у підприємства №1, в якого перебувала в розпорядженні не менше 10 млн.грн, але протягом усього строку, що більш сприятливо.

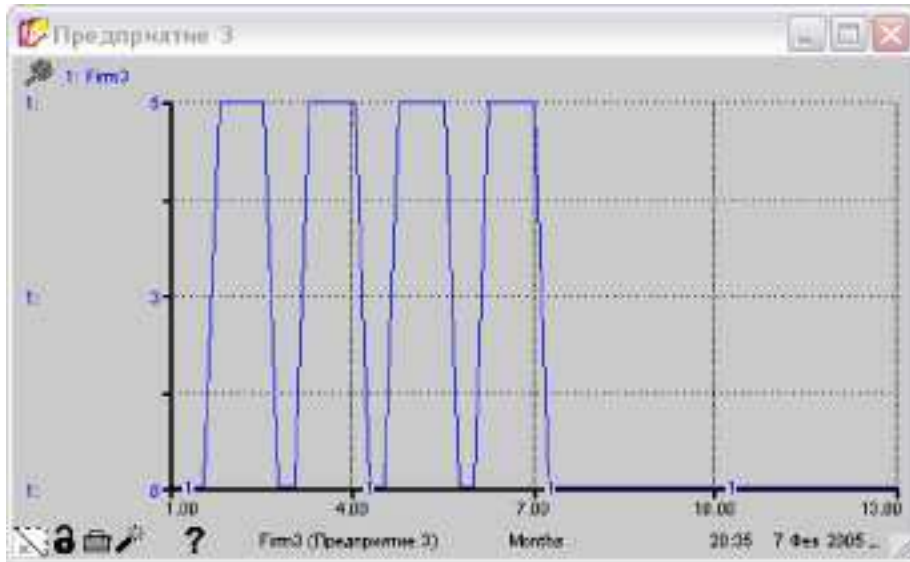


Рис.14.24 Кредитний потік для підприємства №3

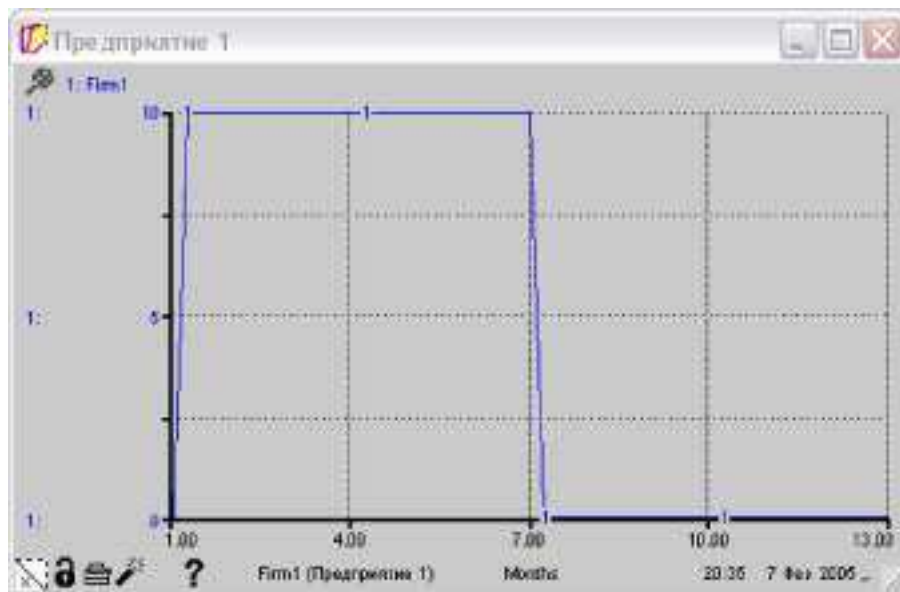


Рис.14.25 Кредитний потік для підприємства №1

Для удосконалення моделі треба в ній врахувати, що банк може знову видавати кошти разом з відсотками, які повертають підприємства, у вигляді кредитів з урахуванням норми обов'язкового резерву.

14.4 Імітаційна модель реалізації інноваційних енергозберігаючих проектів у житлово-комунальному господарстві

Відомо, що зношування основних фондів комунальних підприємств житлово-комунального господарства міста становить 65% і вище при нормативних вимогах – не більше 25%. Таке зношування призводить до того, що підприємства є збитковими, у них не вистачає коштів для капітального ремонту основних фондів і доведення інтегрального показника їхнього зношування до нормативної величини. В цих умовах без залучення інвестицій вирішити цю проблему неможливо. А їх ніхто не виділяє підприємствам, тому що вони через свою збитковість не зможуть їх повернути. Виходить замкнуте коло...

У недавньому минулому на неефективність використання ресурсів в житлово-комунальному господарстві ніхто не звертав уваги, тому що воно дотувалося за рахунок держави. Все змінилося при переході до ринкових умов. Тепер різницю між собівартістю надання житлово-комунальних послуг і оплатою за тарифом покривають міські адміністрації з коштів місцевих бюджетів. Без модернізації основних фондів комунальних підприємств неможливо зменшити це навантаження на місцеві бюджети. Щоб його зменшити, розробляються інвестиційні енергозберігаючі програми і проекти, що вимагають, як правило, значних інвестицій. Зокрема, за участю енергосервісних компаній, на комунальних підприємствах вводять в дію менш енергоємне устаткування.

Рішенням проблеми може бути виділення коштів з бюджету міського розвитку або одержання кредитів державних банків за умови, що вони можуть бути повернуті за рахунок істотної економії енергоресурсів при наданні житлово-комунальних послуг.

Механізм забезпечення такого процесу може бути наступним (**рис.16.26**). Енергосервісна компанія починає комплексні роботи із заміни енергоємного устаткування в інженерних мережах. Заміняються системи управління двигунами насосів та іншим енергоємним устаткуванням, які використовуються при наданні комунальних послуг. Зекономлені ресурси від заміни устаткування можуть бути реінвестовані на капітальний ремонт основних фондів, наприклад, інженерних

мереж. В результаті забезпечується безперервний потік реалізації енергозберігаючих проектів на основі зворотного механізму їхнього рефінансування за допомогою енергосервісних компаній, більшість яких являють собою приватний бізнес. Вони не використовують бюджетні кошти для реалізації енергозберігаючих проектів на комунальних підприємствах міста.

Якби органи місцевого самоврядування направляли бюджетні кошти розвитку в реалізацію проектів енергосервісних компаній, то процес капітального ремонту інженерних мереж міг би здійснюватися значно швидше.

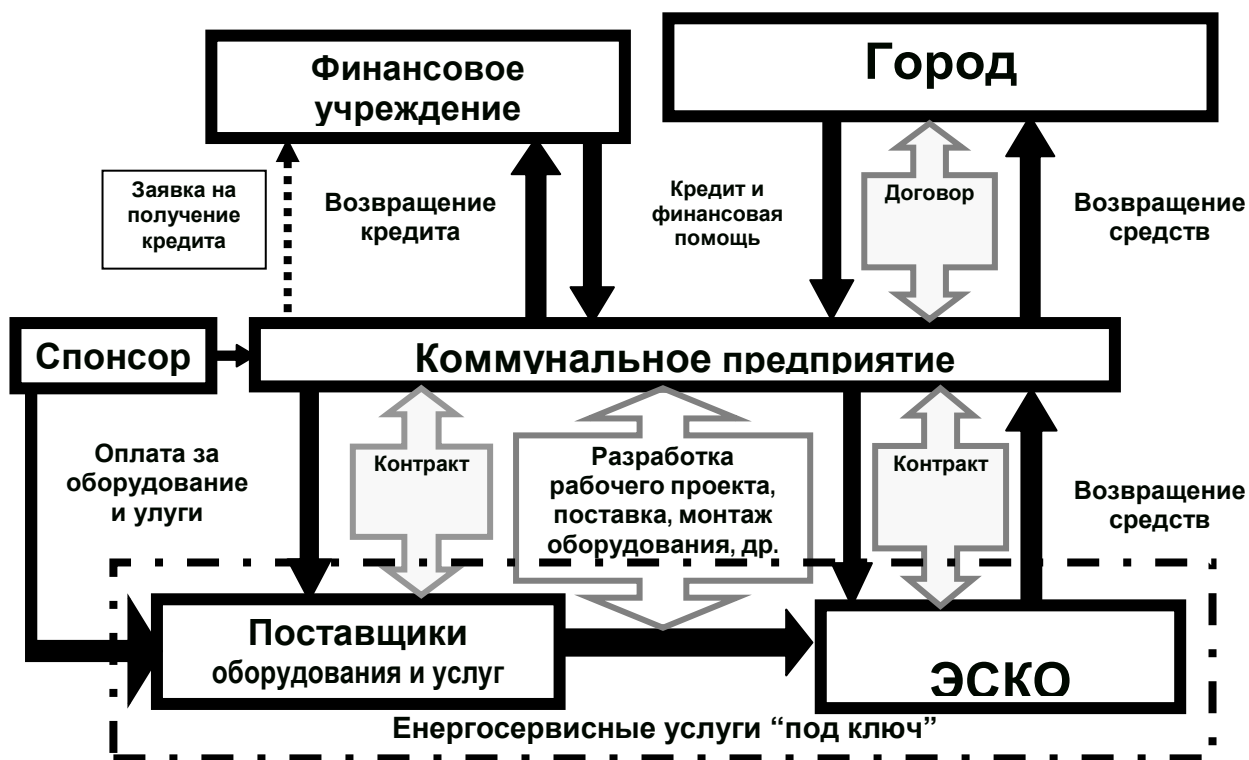


Рис.14.26 Схема зворотного механізму інвестування коштів на енергозберігаючі проекти.

Для цього механізму була в середовищі Vensim 5.0 PLE створена імітаційна модель (рис.14.27) з використанням методу системної динаміки, в основі якого лежить подання про функціонування динамічної системи як сукупності фінансових, ресурсних і людських потоків. Ця модель дозволяє:

- прогнозувати відповідні частини бюджетів міста і комунальних підприємств й енергосервісних компаній,
- оцінювати обсяги енергозбереження, повернення інвестицій в бюджет розвитку міста, витрат на поставку устаткування і на послуги енергосервісних компаній.

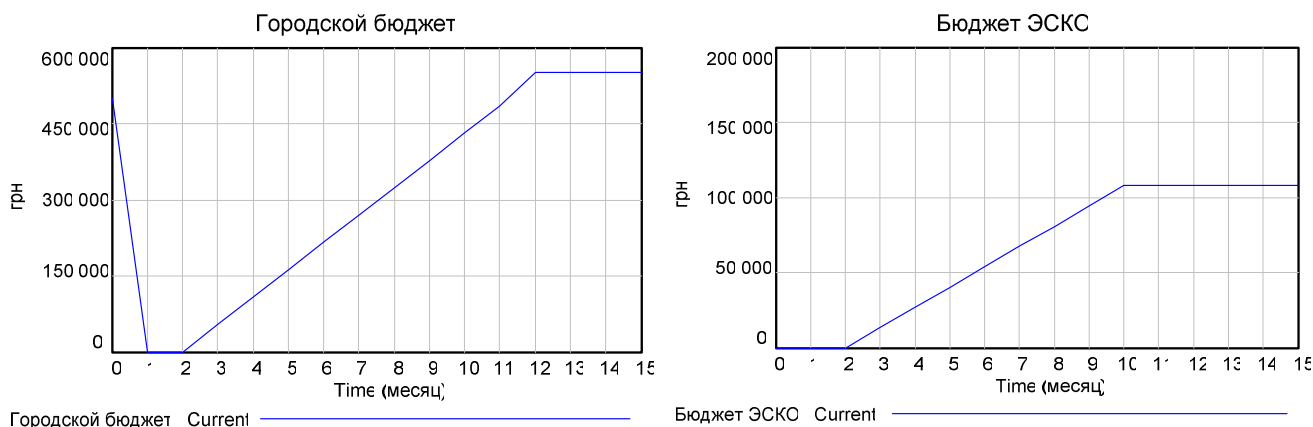


Рис.14.28 Динаміка зміни бюджетів

З діаграми бюджету комунального підприємства на **рис.14.29** видно, що після 12-го місяця воно може починати або новий енергозберігаючий проект, або акумулювати фінансові кошти на капітальний ремонт своїх інженерних мереж. З діаграми темпів повернення інвестицій видно, що з 1-го по 11-й місяць іде активний процес повернення інвестицій за рахунок енергозберігаючих проектів з бюджету комунального підприємства у міський бюджет.

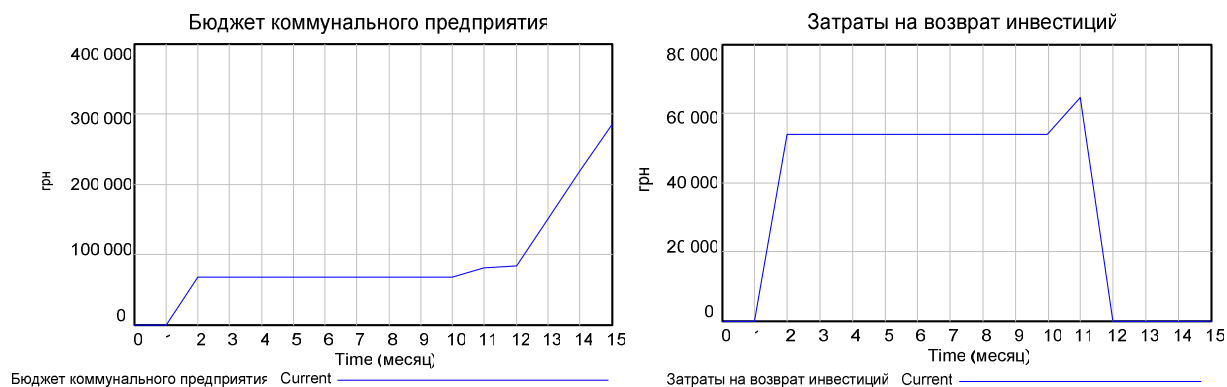


Рис.14.29 Динаміка зміни бюджету комунального підприємства й повернення інвестицій

На даній імітаційній моделі можна провести обчислювальні експерименти з дослідження різних стратегій впровадження енергозбереження на комунальних підприємствах міста за участю енергосервісних компаній.

14.5 Моделювання виведення на ринок інноваційних продуктів

Процес виведення на ринок інноваційних продуктів, що включає в себе їхній маркетинг, запуск і розвиток вимагає чималих інвестицій. Тому виникає необхідність точного прогнозування темпів доходів і придбання продуктів споживачами. Особливо це стосується таких сфер, як послуги широкосмугового мобільного зв'язку, створення і розвиток соціальних мереж в Інтернеті та ін.

Моделі, що описують виведення на ринок нового продукту (дифузійні моделі), вперше з'явилися на початку 1960-х років. Вони характеризували його поширення за допомогою S-образної кривої - моделі Басса, названої так за ім'ям свого розроблювача Фрэнка Басса [56].

Ця модель описувала дифузійну поширення новинок на прикладі споживчих товарів тривалого призначення. Її суть полягає в наступному. На деякий ринок надходить новий продукт (товар або послуга), що не має аналогів і, відповідно, конкуренції з боку інших продуктів. З'являється певна кількість людей, які бажають придбати цей продукт, і людей, які вже зробили його покупку. Частка людей, які купують продукт в момент часу t , визначається функцією розподілу покупок у часі $F(t)$, що залежить від коефіцієнта зовнішнього впливу p (коефіцієнта інновації) і від коефіцієнта внутрішнього впливу q (коефіцієнта імітації).

Модель припускає, що акт покупки відбувається або під впливом реклами і засобів масової інформації (ця категорія покупців називається новаторами), або під впливом людей, які вже зробили покупку (ця категорія покупців називається імітаторами). Другий метод впливу на покупців називається методом «сарафанного радіо». Зовнішній вплив приймається постійним, а внутрішній вплив, що залежить від самої соціальної системи, – пропорційно збільшується в міру росту кількості людей, які вже зробили покупку.

Число покупок $n(t)$, здійснених у момент часу t , дорівнює добутку числа потенційних покупців продукту m на частку покупців $f(t)$, які зробили покупку в момент часу t : $n(t) = mf(t)$.

Аналогічно, $N(t)=mF(t)$ – є загальне число людей, які зробили покупки за весь минулий період. В [56] приводиться диференціальне рівняння, що описує динаміку продажів продукту:

$$\frac{dN(t)}{dt} = \left[p + \frac{q}{m} N(t) \right] [m - N(t)]. \quad (14.1)$$

Його аналітичне рішення, у припущенні сталості параметрів p, q, m в області визначення цієї функції $[0, t]$, виглядає так:

$$N(t) = m \left[\frac{1 - e^{-(p+q)t}}{1 + \frac{q}{p} e^{-(p+q)t}} \right]. \quad (14.2)$$

Таким же чином можна одержати аналітичне рішення для перемінної $n(t)$:

$$n(t) = m \left[\frac{p(p+q)^2 e^{-(p+q)t}}{(p + qe^{-(p+q)t})^2} \right]. \quad (14.3)$$

Модель Басса можна побудувати за допомогою систем імітаційного моделювання. На **рис.14.30** показана модель виведення інноваційних продуктів на ринок, яка була сформована в роботі [58] з використанням програмних засобів **Vensim**. В ній було розглянуто прогностні моделі для ряду категорій продуктів, наведених в **табл.14.1**.

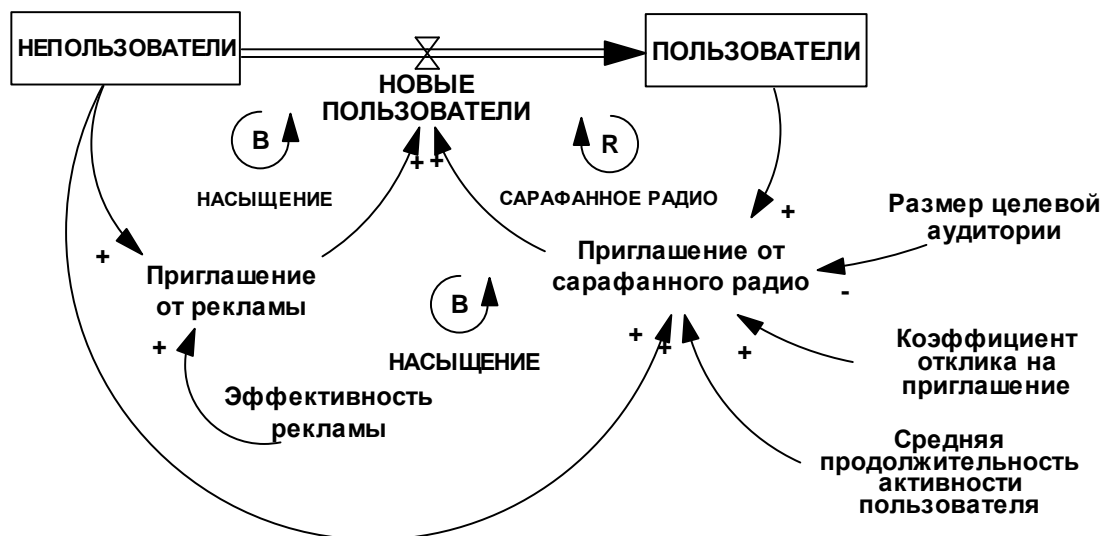


Рис.14.30 Змістовна модель виведення нових продуктів на ринок

Приклади параметрів моделі Басса для ряду продуктів

Продукт	p	q
Телевізори (чорно-білі)	0,028	0,25
Телевізори (кольорові)	0,005	0,84
Кондиціонери	0,010	0,42
Сушарки для одягу	0,017	0,36
Пом'ягчувачі води	0,018	0,30
Магнітофони	0,025	0,65
Щільникові телефони	0,004	1,76
Парові праски	0,029	0,33
Закусочні McDonalds	0,018	0,54
Ковдри з підігрівом	0,006	0,24

Позначення: p – коефіцієнти зовнішнього впливу, q – коефіцієнти внутрішнього впливу.

Вбудований в пакет **Vensim** математичний апарат дозволяє вирішувати системи нелінійних диференціальних рівнянь. Так що необхідність їх аналітичного рішення відпадає.

Потік нових користувачів - новаторів створюється тут за рахунок реклами інноваційного продукту, ефективність якої задається параметром p . Потік нових користувачів - імітаторів створюється за рахунок запрошення від “сарафанного радіо”. Він визначається параметром q , що дорівнює добутку середньої тривалості активності користувача на коефіцієнт відгуку на запрошення.

Результати проведеного комп'ютерного експерименту на цій моделі для щільникових телефонів з параметрами з табл. 14.1 і числом потенційних користувачів $m=1$ млн. людей, наведено на рис.14.31, де за роками показана динаміка накопичувального росту числа користувачів, які купили щільникові телефони за розглянутий період часу. На рис.14.32 показана за роками динаміка зміни потоку нових користувачів, максимум функції якого досягається в точці перегину $T^* = 3,63$ роки, що обчислюється за формулою:

$$T^* = \frac{1}{p+q} \ln\left(\frac{q}{p}\right). \quad (14.4)$$

З графіка рис.14.32 видно, що функція $n(t)$, яка описує число покупок у кожний момент часу, має максимум у точці T^* . Умовою існування максимуму

функції $n(t)$, і, отже, точки перегину у функції $N(t)$, є $p < q$. В протилежному випадку функція $n(t)$ є монотонно убутною функцією.

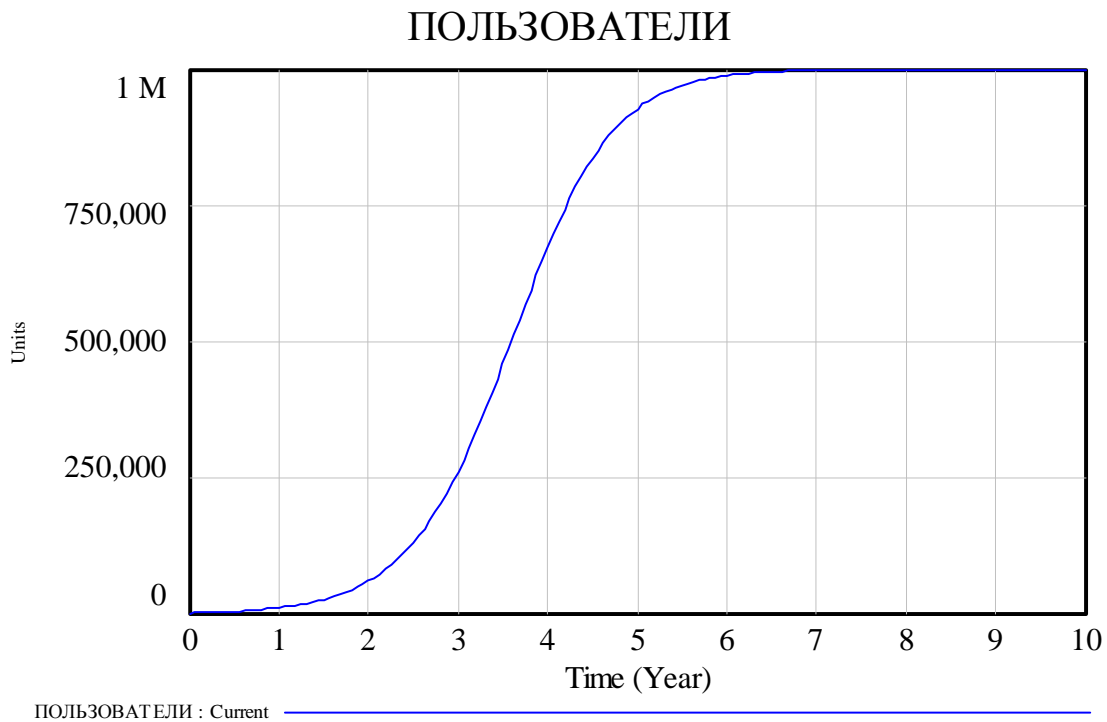


Рис.14.31 Накопичувальний за роками ріст числа користувачів $N(t)$, які купили щільникові телефони



Рис.14.32 Динаміка зміни за роками числа $n(t)$ нових користувачів, які купили щільникові сотові телефони

Лістинг програми в Vensim

- (01) FINAL TIME = 10
Units: Year
The final time for the simulation.
- (02) INITIAL TIME = 0
Units: Year
The initial time for the simulation.
- (03) SAVERPER = TIME STEP
Units: Year
The frequency with which output is stored.
- (04) TIME STEP = 0.0625
Units: Year
The time step for the simulation.
- (05) Коефіцієнт відгуку на запрошення = 0.0176
Units: Dimensionless
- (06) НЕКОРИСТУВАЧІ= INTEG (-НОВІ КОРИСТУВАЧІ,
Розмір цільової аудиторії - КОРИСТУВАЧІ)
Units: Units
- (07) НОВІ КОРИСТУВАЧІ=
Запрошення від реклами + Запрошення від сарафанного радіо
Units: Units/Year
- (08) КОРИСТУВАЧІ= INTEG (НОВІ КОРИСТУВАЧІ, 0)
Units: Units
Число користувачів у системі.
- (09) Запрошення від реклами = Ефективність реклами* НЕКОРИСТУВАЧІ
Units: Units/Year
- (10) Запрошення від сарафанного радіо =
Середня тривалість активності користувача
Коефіцієнт відгуку на запрошення НЕКОРИСТУВАЧІ
*КОРИСТУВАЧІ/ Розмір цільової аудиторії
Units: Units/Year
- (11) Розмір цільової аудиторії=1e+006
Units: Units
Розмір цільової аудиторії.
- (12) Середня тривалість активності користувача =100
Units: 1/Year
- (13) Ефективність реклами = 0.004
Units: 1/Year

14.6 Моделювання модернізації мереж водопостачання

Навколо інвестицій в модернізацію інженерної інфраструктури водопостачання і водовідведення склалась така ситуація. В більшості підприємств водопровідно-каналізаційного господарства і органів місцевого самоврядування не вистачає власних можливостей фінансування модернізації. А залучення зовнішніх джерел інвестування проблематично через відсутність чітких «правил гри» для приватних інвесторів і твердого директивного регулювання тарифів і умов оплати послуг. Одним із напрямків вирішення цієї проблеми є реалізація в даному господарстві енергозберігаючих проектів. Для того, щоб досліджувати ці можливості модернізації інженерних мереж водопостачання для зміни їхнього стану, була розроблена імітаційна модель [51] за допомогою програмного засобу **Vensim**, представлена на **рис.14.33**.

В цій моделі виділено три види стану інженерних мереж водопостачання: нові мережі, наявні нормальні мережі і зношені мережі. Було враховано, що побудовані нові мережі водопостачання внаслідок зношування поступово переходять у категорію нормальних мереж, які в свою чергу, згодом стають зношеними мережами, що вимагають капітального ремонту, після чого вони стають знову нормальними мережами водопостачання.

У моделі передбачається, що зекономлені за рахунок енергозбереження фінансові ресурси направляються на капітальний ремонт зношених мереж водопостачання, фінансований з міського бюджету. В якості мірила енергозбереження використовується показник питомого видатку електроенергії на одиницю об'єму подаваної споживачам води. Цей показник у країнах ЄС становить 0,6 кВт-год/м³. У водопровідному господарстві міста Харкова цей показник становить 1 квт-годин/м³ [51]. Перемінна “частка енергозбереження в послугах водопостачання” відображає хід водопостачання на комунальних підприємствах м. Харкова.

При імітаційному моделюванні досліджувалися можливості підвищення енергоефективності за рахунок зниження питомого видатку електроенергії в системі водопостачання м. Харкова при реалізації енергозберігаючих проектів на 40%.

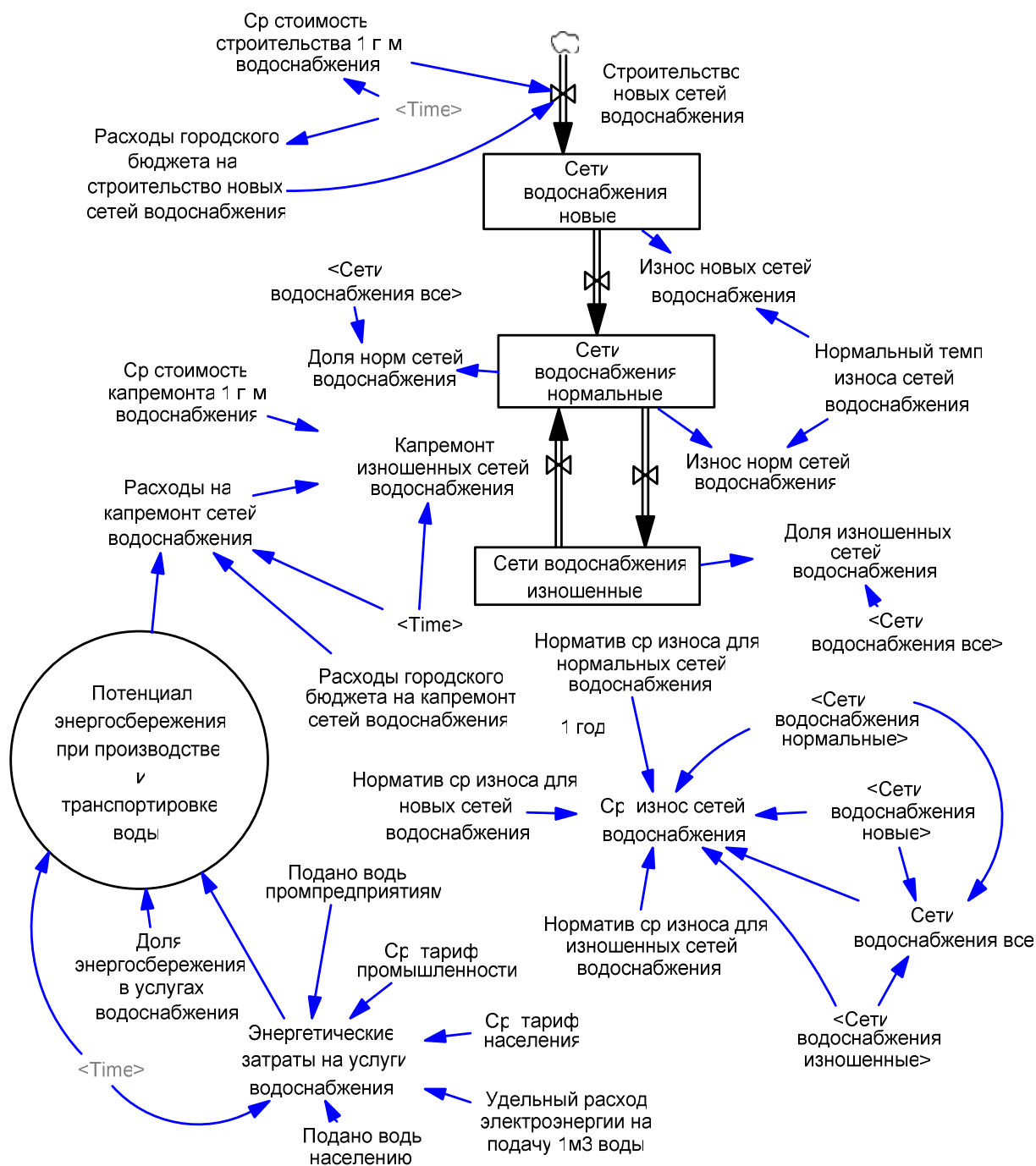


Рис.14.33 Імітаційна модель модернізації інженерних мереж водопостачання міста

На рис.14.34, 14.35 наведені результати комп'ютерного експерименту, проведеного на даній імітаційній моделі: зміна потенціалу енергозбереження в системі водопостачання і зміна видатків на капітальний ремонт її інженерних мереж.

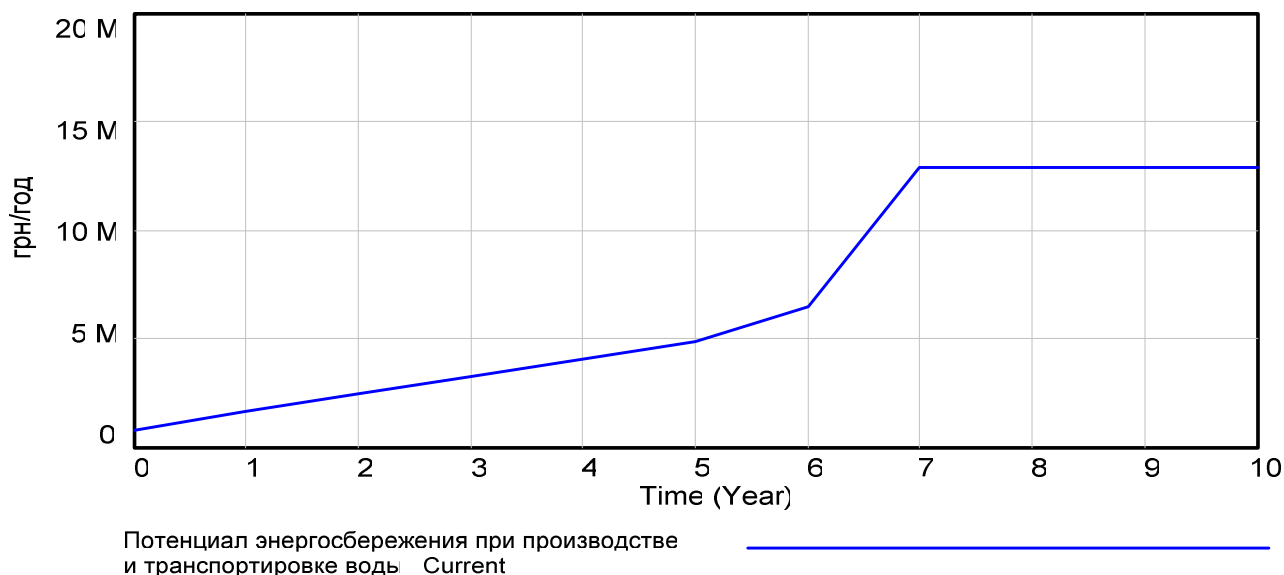


Рис.14.34 Потенціал енергозбереження системи водопостачання

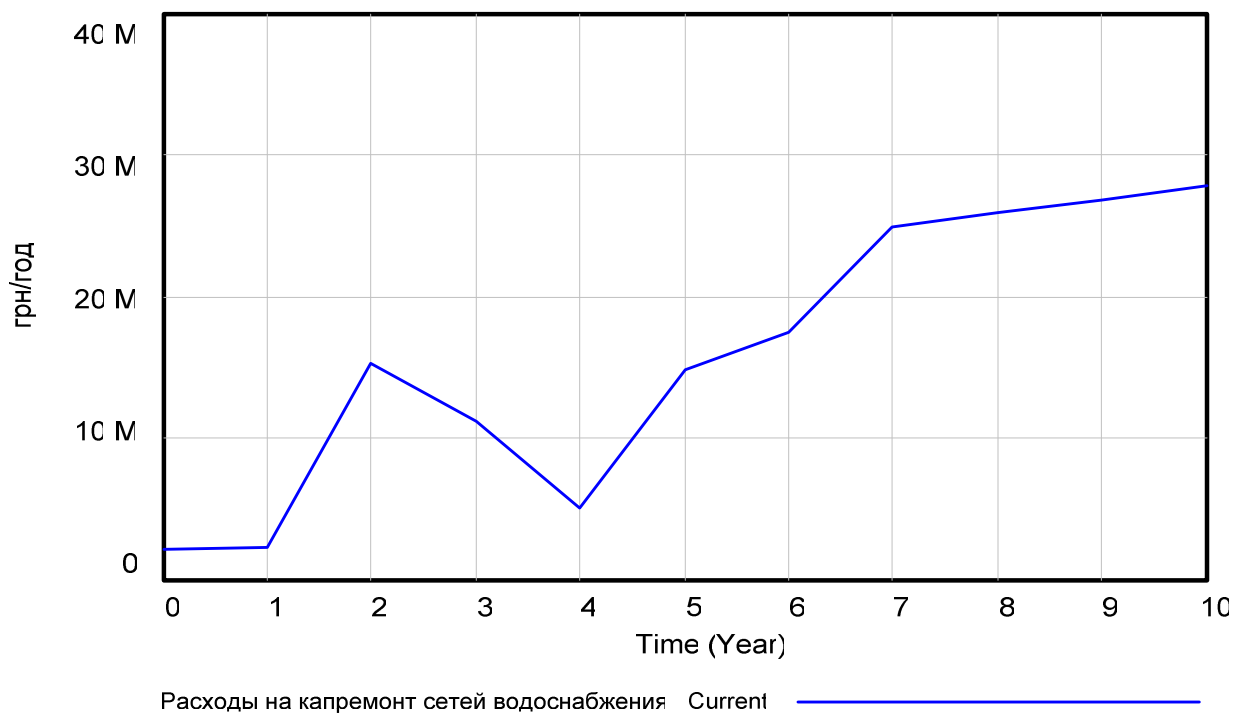


Рис.14.35 Видатки на капітальний ремонт мереж водопостачання

Потенціал енергозбереження відображає поетапний процес підготовки й реалізації енергозберігаючих проектів на комунальних підприємствах водопостачання м. Харкова (рядок 07 у наведеному нижче лістингу програми). Із графіка видно, що вихід на запланований показник питомого видатку електроенергії здійснюється тільки починаючи з сьомого періоду моделювання. Як правило, строк окупності проектів енергозбереження на комунальних підприємствах водопостачання

коливається в межах від півроку до року. Проведений експеримент показав також, що потенціал енергозбереження при виробництві й транспортуванні води споживачам, що направляється на модернізацію інженерних мереж, може бути порівняно з видатками міського бюджету, виділених для цих же цілей.

На **рис.14.36, 14.37** показана зміна станів нормальних і зношених мереж водопостачання. На початковий період моделювання (2000-й рік) середній відсоток зношування мереж становив 58%. Для прийнятої динаміки фінансування модернізації мережі водопостачання ситуація почне незначно поліпшуватися тільки з п'ятого року періоду моделювання, що свідчить про його недостатність. Очевидно, що без збільшення фінансування модернізації інженерних мереж водопостачання не доводиться очікувати підвищення надійності надання послуг водопостачання.

Використовуючи запропоновану модель, можна оцінити реалізації різних стратегій впровадження енергозберігаючих проектів на комунальних підприємствах водопостачання. Для цього необхідно змінити показник динаміки впровадження енергозбереження.

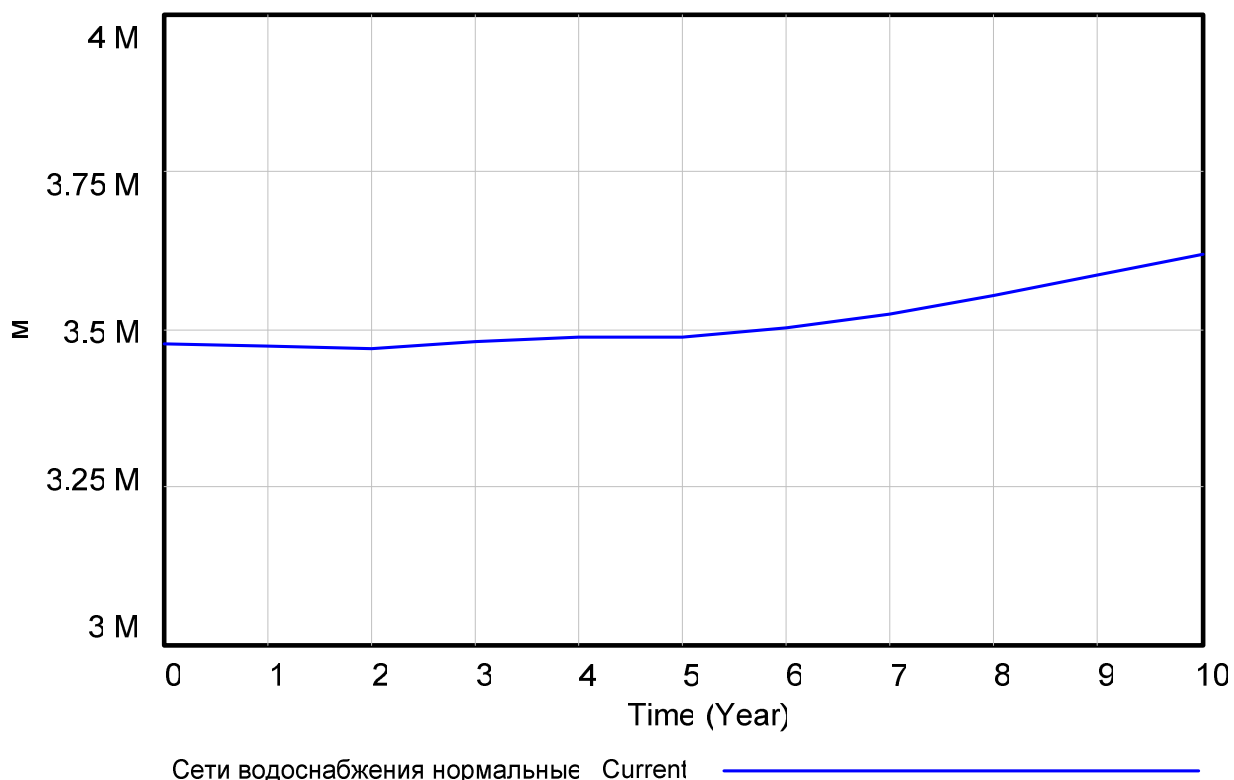


Рис.14.36 Мережі водопостачання нормальні

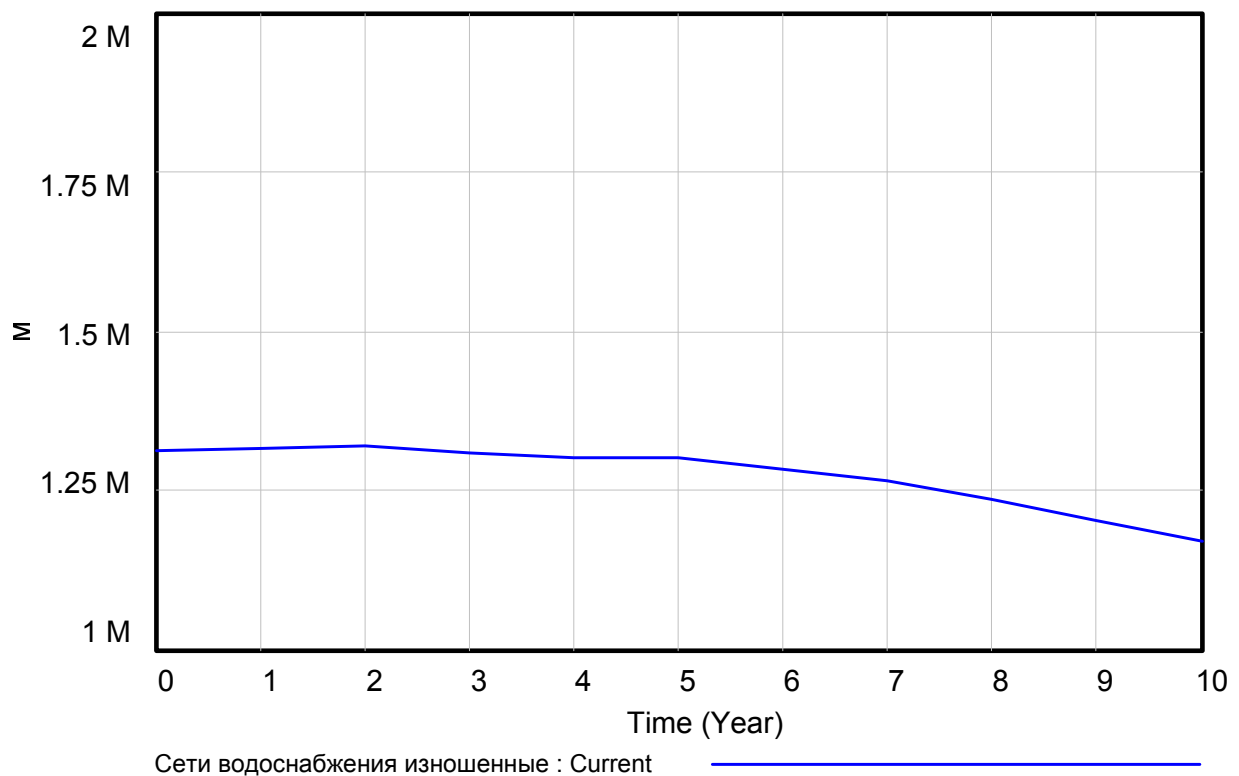


Рис.14.37 Мережі водопостачання зношені

Лістинг програми й використані вихідні дані

- (01) FINAL TIME = 10
Units: Year
The final time for the simulation.
- (02) INITIAL TIME = 0
Units: Year
The initial time for the simulation.
- (03) SAVEPER = TIME STEP
Units: Year [0,?]
The frequency with which output is stored.
- (04) TIME STEP = 1
Units: Year [0,?]
The time step for the simulation.
- (05) Частка зношених мереж водопостачання =Мережі водопостачання зношені/Мережі водопостачання всі. Units: Dmnl
- (06) Частка норм мереж водопостачання =Мережі водопостачання нормальні/Мережі водопостачання всі. Units: Dmnl
- (07) Частка енергозбереження в послугах водопостачання
[(0,0)-(10,10)],(0,0.025), (1,0.05),(2,0.075),(3,0.1),(4,0.125),(5,0.15),
(6,0.2),(7,0.4),(8,0.4),(9,0.4),(10,0.4)). Units: Dmnl
- (08) Зношування нових мереж водопостачання = INTEGER(Мережі водопостачання нові*Нормальний темп зношування мереж водопостачання)
Units: Dmnl

(09) Зношування норм мереж водопостачання = INTEGER(Мережі водопостачання нормальні* Нормальний темп зношування мереж водопостачання). Units: Dmnl

(10) Капремонт зношених мереж водопостачання = INTEGER(Видатки на капремонт мереж водопостачання /Сер вартість капремонту 1 п.м. водопостачання (Time)). Units: м

(11) Нормальний темп зношування мереж водопостачання = 0.002
Units: Dmnl

(12) Норматив порівн зношування для зношених мереж водопостачання=0.8. Units: Dmnl

(13) Норматив порівн зношування для нових мереж водопостачання=0.2
Units: Dmnl

(14) Норматив порівн зношування для нових теплових мереж=0.2
Units: Dmnl

(15) Норматив порівн зношування для нормальних мереж водопостачання=0.5. Units: Dmnl

(16) Подано води населенню ([(0,0)-(10,2e+008)],(0,1.82574e+008), (1,1.67869e+008), (2,1.64e+008),(3,1.47e+008), (4,1.47e+008),(5,1.47e+008), (6,1.47e+008),(7,1.47e+008),(8,1.47e+008),(9,1.47e+008),(10,1.47e+008)). Units: м3

(17) Подано води промпідприємствам

(18) ([(0,0)-(10,2e+008)],(0,8.30546e+007), (1,9.916e+007),(2,1.013e+008),(3,1.139e+008),(4,1.139e+008), (5,1.139e+008), (6,1.139e+008), (7,1.139e+008),(8,1.139e+008), (9,1.139e+008),(10,1.139e+008))
Units: м3

(18) Потенціал енергозбереження при виробництві й транспортуванні води =Енергетичні витрати на послуги водопостачання

*Частка енергозбереження в послугах водопостачання (Time)

Units: грн. /рік

(19) Видатки міського бюджету на капремонт мереж водопостачання ([(0,0)-10,2e+008)],(0,1.327e+006),(1,581000),(2,1.28e+007), (3,8e+006),(4,1e+006),(5,1e+007),(6,1.1e+007),(7,1.2e+007),(1,3e+007), (9,1.4e+007),(10,1.5e+007)). Units: грн. /рік

(20) Видатки міського бюджету на будівництво нових мереж водопостачання = WITH LOOKUP (Time,([(0,0)-(10,10)],(0,0),(5,0),(10,0)))

Units: грн/Year

(21) Видатки на капремонт мереж водопостачання = Потенціал енергозбереження при виробництві й транспортуванні води +Видатки міського бюджету на капремонт мереж водопостачання (Time). Units: грн/рік

(22) Мережі водопостачання всі = Мережі водопостачання нові + Мережі водопостачання нормальні +Мережі водопостачання зношені
Units: м

(23) Мережі водопостачання зношені = INTEG (+Зношування норм мереж водопостачання - Капремонт зношених мереж водопостачання, 1.312e+006). Units: м

(24) Мережі водопостачання нові= INTEG (+Будівництво нових мереж водопостачання - Зношування нових мереж водопостачання, 10000). Units: м

(25) Мережі водопостачання нормальні = INTEG (Капремонт зношених мереж водопостачання + Зношування нових мереж водопостачання - Зношування норм мереж водопостачання, 3.4754e+006). Units: м

(26) Ср зношування мереж водопостачання =(Мережі водопостачання нормальні *Норматив порівн зношування для нормальних мереж водопостачання + Мережі водопостачання нові *Норматив порівн зношування для нових мереж водопостачання + Мережі водопостачання зношені *Норматив порівн зношування для зношених мереж водопостачання)/Мережі водопостачання всі. Units: Dmnl

(27) "Ср вартість капремонту 1 п.м. водопостачання"
([(0,0)-(10,1000)],(0,663),(1,700),(2,800),(3,800),(4,663),(5,663),(6,663),
(7,663),(8,663),(9,663),(10,663)). Units: грн/м

(28) "Ср вартість будівництва 1 п.м. водопостачання" = WITH LOOKUP
Time, ([(0,0)-(10,1000)],(0,800),(5,900),(10,1000)). Units: грн/м

(29) Ср тариф населення=0.11. Units: грн/квтч

(30) Ср тариф промисловості=0.14. Units: грн/квтч

(31) Будівництво нових мереж водопостачання =Видатки міського бюджету на будівництво нових мереж водопостачання /"Ср вартість будівництва 1 п.м. водопостачання". Units: м

(32) Питомий видаток електроенергії на подачу 1м3 води =1.
Units: квтч/м3

(33) Енергетичні витрати на послуги водопостачання =(Подано води населенню (Time)* Ср тариф населення +Подано води промпідприємств (Time) *Ср тариф промисловості)*Питомий видаток електроенергії на подачу 1м3 води. Units: грн

Розділ 15

Моделювання динаміки надходження заявок на будівництво і їх обслуговування

15.1 Опис процедури обслуговування забудовників

Одним із джерел інформації для аналізу процедури подачі заявок і їхнього обслуговування у відділі початкових даних є web-сайт Харківської міськради, де розміщено відомості по рішеннях сесій міської ради про надання земельних ділянок для будівництва (реконструкції) об'єктів і дозволів їх відводу забудовникам [69]. На базі цих даних сформована **табл. 15.1**. Було виявлено, що для значної кількості заявок рішення не приймаються внаслідок зміни планів або відмови, якщо порушено строк, що надається на розробку проекту відводу земельної ділянки.

Таблиця 15.1

Кількість ухвалених рішень щодо заявок на будівництво

Рік/ квартал сесії	Кіл-сть дозволів на проектування відводу ділянок	Кіл-сть рішень про надання ділянок	Загальна кількість рішень за заявками
1/1	55	120	175
1/2	64	75	139
1/3	61	79	140
1/4	72	84	156
За 1-й рік			610
2/1	48	89	137
2/2	88	61	149
2/3	66	68	134
2/4	89	138	227
За 2-й рік			647
3/1	62	101	163
3/2	78	111	189
3/3	80	124	204
3/4	101	202	303
За 3-й рік			859
Усього	894	1312	2206

Відділ підготовки початкових даних займається підготовкою проектів рішень міськради і його виконкому з питань містобудування. Близько 70% цих проектів пов'язана з одержанням дозволів на будівництво [69]. Працівниками відділу вручну ведеться поточний архів пакетів документів, які надходять від замовників. Крім цього, займає багато часу роботи відділу співробітництво із МВС, прокуратурою та іншими органами щодо підготовки необхідних даних по забудовникам. Співробітники зайняті також у комісіях, де розглядаються різноманітні питання про самовільне будівництво, заявки та ін.

Найбільш трудомістка частина роботи пов'язана з прийомом і консультуванням забудовників, з формуванням і обробкою пакетів документів для підготовки проектів рішень. Відділ часто є єдиним місцем, куди забудовник може звернутися за консультацією. Фахівці відділу працюють безпосередньо з ними, приймаючи в них повний пакет документів. Чимало часу йде на підготовку вже використаних пакетів документів для здачі їх в архів (у відділі залишаються деякі основні дані). Два рази на тиждень половина робочого дня витрачається на прийом громадян.

Для підготовки проекту рішення співробітникам відділу необхідно перевірити наявність всіх необхідних документів і завізувати його в юридичному відділі. До складу пакета документів для підготовки проекту рішення про згоду на проектування відводу земельної ділянки входять:

- архітектурне завдання на розробку містобудівного обґрунтування розміщення об'єкта;
- акт установлення й узгодження границь земельної ділянки і акт вибору ділянки для проектування й будівництва об'єкта з листом його узгодження;
- містобудівне обґрунтування розміщення об'єкта;
- звертання до мера про можливість проектування і будівництва з розробкою проекту відводу земельної ділянки;
- листа погоджень містобудівного обґрунтування з Харківською міською санітарно-епідеміологічною станцією (СЕС), з Управлінням екології і природних ресурсів у Харківській області, з Управлінням земельних відносин, з депутатом Харківської міської ради;
- комплексний висновок Головного управління містобудування, архітектури і земельних відносин.

Далі треба безпосереднє складання проекту рішення і його здача наприкінці тижня в канцелярію міськради.

Аналогічна робота ведеться при підготовці проекту рішення про надання земельної ділянки. До складу пакета документів входять:

- заява про надання земельної ділянки для будівництва й експлуатації на ім'я Харківського міського голови;
- технічне завдання на виконання робіт зі складання проекту відводу земельної ділянки;
- проект рішення про надання згоди на розробку проекту відводу земельної ділянки;
- проект відводу земельної ділянки;
- листа погоджень проекту відводу земельної ділянки з управлінням культури обласної адміністрації, з міською СЕС, з обласним управлінням екології і природних ресурсів, з міським управлінням земельних ресурсів;
- висновок державної землевпорядної експертизи обласного головного управління земельних ресурсів і управління містобудування та архітектури.

При проведенні спостережень за роботою співробітника відділу із заявками було отримано дані, наведені в **табл.15.2**. Показана структура щоденних витрат робочого часу співробітника відділу за видами операцій. До інших робіт відноситься підготовка різної інформації з вимогою МВС, прокуратури, інших контрольно-ревізійних органів, ведення та обробка поточного архіву пакетів документів, які подаються забудовниками.

Фонд робочого часу відділу по обслуговуванню забудовників за 2 місяці (проміжок між сесіями міськради) було визначено, виходячи із середнього часу обслуговування однієї заявки в 3,17 людино-годин, і наявності 6 співробітників відділу. Він становить 1620 людино-годин. Час на підготовку всього обсягу робіт по 506 проектам рішень дорівнює 1603 людино-годинам.

Таблиця 15.2

Витрати часу на одну заявку, годин

Операції	Витрати часу	
	На заявку	На співробітника в день
Консультація замовника, перевірка і візування пакета документів	2,5	4,5
Формування і передача проекту рішення	0,67	1,5
Всього на безпосереднє обслуговування заявок	3,17	6,0
Інші роботи		2,0
Всього за день		8,0

15.2 Прогнозування потоку заявок

Для прогнозування кількості заявок використана наступна модель:

$$k = (y + e) \times s, \quad (15.1)$$

де y – значення тренда, e – помилка, s - індекс сезонності.

Тренд виділяється за допомогою згладжування тимчасового ряду, яким у цьому випадку є наведена в **табл. 15.1** поквартальна інформація про кількість прийнятих на сесії міськради рішень за заявками на будівництво і проектування. Одночасно визначається індекс сезонності. Потім, застосовуючи метод найменших квадратів до отриманих значень y , формується рівняння регресійної кривої, що дозволяє виконувати прогноз кількості заявок у наступному періоді часу. Процес згладжування ряду, виконаний методом слизької середньої по чотирьох точках [44], наведено в **табл.15.3**. За цим методом спочатку формуються підсумки по кількості рішень за перші чотири квартали 1-го року. Далі робиться здвиг на один квартал і підсумки за квартали 2, 3, 4 (1-го року) і 1-го кварталу 2-го року. Потім - 3, 4, 1, 2, і т.д. Середні квартальні значення отримують діленням підсумкових сум на 4. Центровані середні значення y формуються для сусідніх значень середніх. Після цього визначається індекс сезонності за формулою:

$$s = k / (y + e). \quad (15.2)$$

Оскільки отримані оцінки індексу сезонності включають помилку, то в **табл. 15.4** виконано його корегування й отримані значення, занесені в стовпець s_j^* **табл. 15.3**.

Таблиця 15.3

Процес згладжування ряду

x	Рік /кВ	k	За 4 кВ	Порівн за кВ	y_i	s	s_j^*	$y+e (y_i^*)$	y	$ y_i - y / y_i$
1	1/1	175					0,93	189,3	180,4	0,047
2	1/2	139					0,96	144,2	160,0	0,110
3	1/3	140					0,90	156,4	147,2	0,059
4	1/4	156	610	152,5			1,22	128,2	141,2	0,101
5	2/1	137	572	143	147,8	0,95	0,93	148,2	141,3	0,047
6	2/2	149	582	145,5	144,3	1,08	0,96	154,6	146,6	0,051
7	2/3	134	576	144	144,8	0,95	0,90	149,7	156,5	0,045
8	2/4	227	647	161,8	152,9	0,98	1,22	186,6	170,1	0,089
9	3/1	163	673	168,3	165	0,81	0,93	176,3	186,6	0,059
10	3/2	189	713	178,3	173,3	1,31	0,96	196,1	205,3	0,047
11	3/3	204	783	195,8	187	0,87	0,90	227,9	225,4	0,011
12	3/4	303	859	214,8	205,3	0,92	1,22	249,1	246,0	0,012
									2106,5	0,68

Тут величина індексу сезонності поставлена в залежність від номера кварталу. Значення тренду y , знайдені за формулою $(y+e)=k/s$, подано в наступному стовпці **табл.15.3**.

Таблиця 15.4

Розрахунок індекса сезонності

Квартал	s_{j1}	s_{j2}	Середній s_j	s_j^*
1	0,948	0,872	0,909	0,925
2	0,975	0,921	0,948	0,964
3	0,948	0,812	0,880	0,895
4	1,081	1,310	1,196	1,216

Умовою формування рівняння регресійної кривої (лінії тренда) є мінімум суми квадратів відхилень її ординат від ординат точок, виконання якого приводить до системи рівнянь:

$$\begin{cases} a_0 n + a_1 \sum x_i + a_2 \sum x_i^2 + a_3 \sum x_i^3 = \sum y_i \\ a_0 \sum x_i + a_1 \sum x_i^2 + a_2 \sum x_i^3 + a_3 \sum x_i^4 = \sum x_i y_i \\ a_0 \sum x_i^2 + a_1 \sum x_i^3 + a_2 \sum x_i^4 + a_3 \sum x_i^5 = \sum x_i^2 y_i \\ a_0 \sum x_i^3 + a_1 \sum x_i^4 + a_2 \sum x_i^5 + a_3 \sum x_i^6 = \sum x_i^3 y_i \end{cases} \quad (15.3)$$

Рішення цієї системи зроблено за допомогою MS Excel. Знайдені значення коефіцієнтів a_i і сформоване рівняння регресійної кривої зображено на **рис.15.1**.

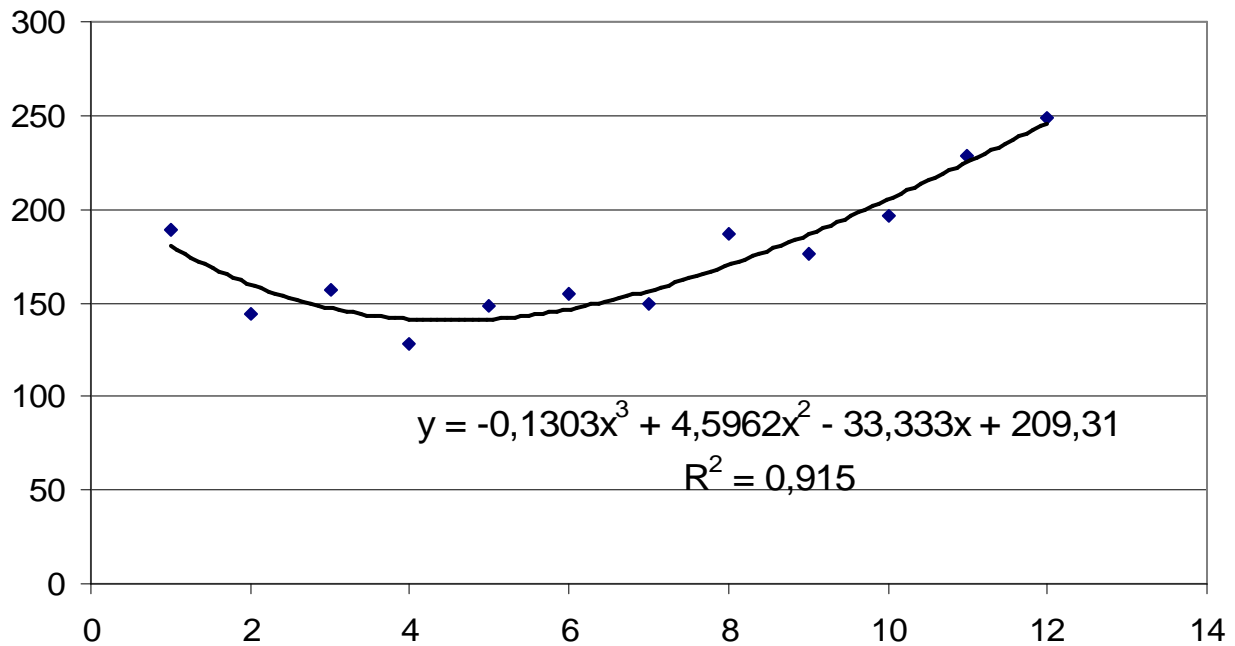


Рис.15.1 Рівняння тренду

Середнє значення помилки апроксимації становить $(0,68/12) \times 100\% = 5,65\%$ (див. три останні стовпці **табл.17.3**). Воно знайдене за формулою [44]:

$$e = \frac{1}{n} \sum \left| \frac{y - y_i}{y_i} \right| \cdot 100 \% \quad (15.4)$$

При значенні коефіцієнта детермінації $R^2 = 0,915$ коефіцієнт кореляції буде дорівнювати: $r = \sqrt{R^2} = 0.96$. Таким чином, достовірність моделі дорівнює 96%. Очікувана кількість заявок k в 4-му році, визначена за допомогою формули (15.1), наведена в **табл.15.5**.

Таблиця 15.5

Прогноз на 4-й рік

Рік/ кв.	x	y	s	k
4/1	13	266,5	0,93	248
4/2	14	286,0	0,96	276
4/3	15	303,7	0,90	272
4/4	16	318,9	1,22	388
Всього заявок				1182

Таким чином, можливе число заявок на будівництво зросте в 4-му році в $1182/859 = 1,34$ рази. На основі отриманих даних прогнозу можна розробити план їхнього обслуговування.

15.3 Умови проведення дослідження

Це дослідження необхідно для виявлення можливостей прискорення одержання дозволів на будівництво і, як наслідок, прискорення збільшення робочих місць, що, в свою чергу, приводить до збільшення платоспроможності населення і росту надходжень в бюджет за рахунок податків і соціальних відрахувань. Крім того, збільшення числа обслужених заявок, у випадку оплати послуг, збільшить дохід міськвиконкому.

Для прискорення руху заявок при виробленні рішень в конкретних відділах треба скоротити час очікування в черзі на обслуговування, а для цього необхідно збільшити кількість співробітників відділів. Це потребує додаткових витрат, які можуть бути компенсовані зниженням збитку від очікування в черзі обслуговування. Таким чином, треба знайти компромісне рішення щодо кількості працівників, при якому буде мінімальною сума витрат на забезпечення діяльності цих працівників і збитків, що виникають через очікування в черзі на обслуговування.

Для проведення дослідження була розроблена початкова модель з використанням понять масового обслуговування. Ця модель описує виникнення на вході в підрозділи міськвиконкому в деякі моменти часу (випадкові або детерміновані) заявок на обслуговування, проходження ними черги й процес обслуговування заявок, після якого вони залишають систему.

Описуваний процес характеризується числом заявок на вході, режимом їхнього надходження в систему обслуговування, поведженням забудовників. В даному процесі заздалегідь невідомо, скільки заявок надійде в той або інший день. Їх число в одиницю часу можна оцінити за допомогою наступного дискретного пуасонівського розподілу [4]:

$$p(x) = (e^{-\lambda} \times \lambda^x) / x!, \quad (15.5)$$

де $p(x)$ – вірогідність надходження x заявок в одиницю часу;

x – число заявок в одиницю часу, $x = 0, 1, \dots$;

λ – середнє число надходження заявок в одиницю часу (темп).

Відповідні значення вірогідності $p(x)$ визначаються за допомогою таблиці пуасонівського розподілу. Наприклад, при середньому темпі надходження заявок у годину – $x = 2$, вірогідність відсутності заявок на вході в систему протягом години дорівнює 0,135, вірогідність появи однієї або двох заявок - 0,27, трьох – 0,18, чотирьох - близько 0,09, а дев'ять або більше – близька до нуля. Тому що на практиці вірогідність появи заявок не завжди підкоряється пуасонівському розподілу, то потрібно обґрунтувати можливість його використання [4].

Побудова моделі ґрунтується на припущенні, що поведження клієнтів є стандартним: вони стають у чергу, чекають доти, поки не будуть обслужені, тобто вони не кидають чергу і не переходять із однієї черги в іншу. Така модель називається моделлю з очікуванням. Черга має довжину, що може бути обмежена. Коли довжина черги досягає свого максимального розміру, то наступній заявці дається відмова допуску в систему. Процес обслуговування характеризується також числом каналів і фаз обслуговування, що визначають конфігурацію системи, і режимом обслуговування. В даній моделі число каналів обслуговування визначає кількість забудовників, що обслуговуються одночасно. **Число фаз** обслуговування визначається кількістю послідовних **етапів** обслуговування одного забудовника. У відділі розгляду заявок працюють 6 чоловік, які можуть одночасно обслуговувати 6 забудовників.

Така система обслуговування називається **багатоканальною системою** з очікуванням. Її схема показана нижче на **рис. 15.2**. Було прийнято, що в досліджуваній системі вхідний і вихідний потоки заявок мають пуасонівські інтенсивності λ і μ , відповідно. В даній системі є r каналів, які паралельно обслуговують не більше r_m заявок забудовників. Відома середня тривалість обслуговування однієї заявки.

Для проведення дослідження була створена програма на базі СУБД MS Visual Foxpro, в яку вводяться дані по потоках заявок, часу обробки однієї заявки, числу каналів обслуговування (кількість співробітників), суми постійних витрат на співробітника. Вікно результатів моделювання системи приведено на **рис.15.3**.

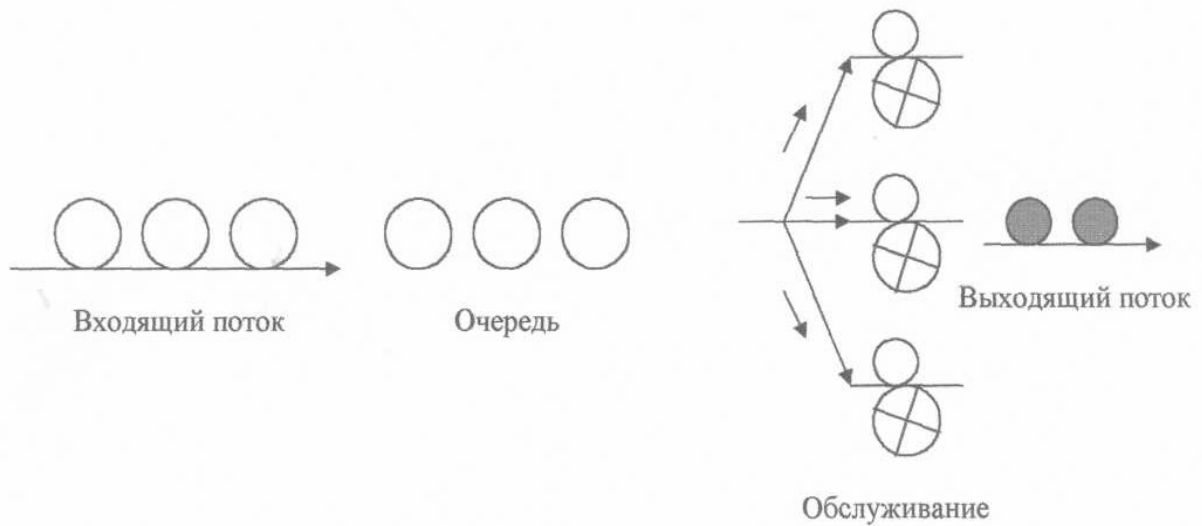


Рис. 15.2 Схема багатоканальної системи обслуговування

Для дослідження залежності витрат від інтенсивності потоку заявок, числа каналів і часу обслуговування за допомогою програми багаторазово проводяться розрахунки вірогідності очікування заявки в черзі, середнього числа заявок у черзі на обслуговування, середнього числа заявок, які перебувають у системі, часу знаходження заявки в черзі, середньої тривалості перебування заявки в системі. Розраховуються також постійні витрати на утримання працівників відділу, коефіцієнт перемінних витрат і витрати, пов'язані з очікуванням (сума недоотриманих доходів за певний відрізок часу).

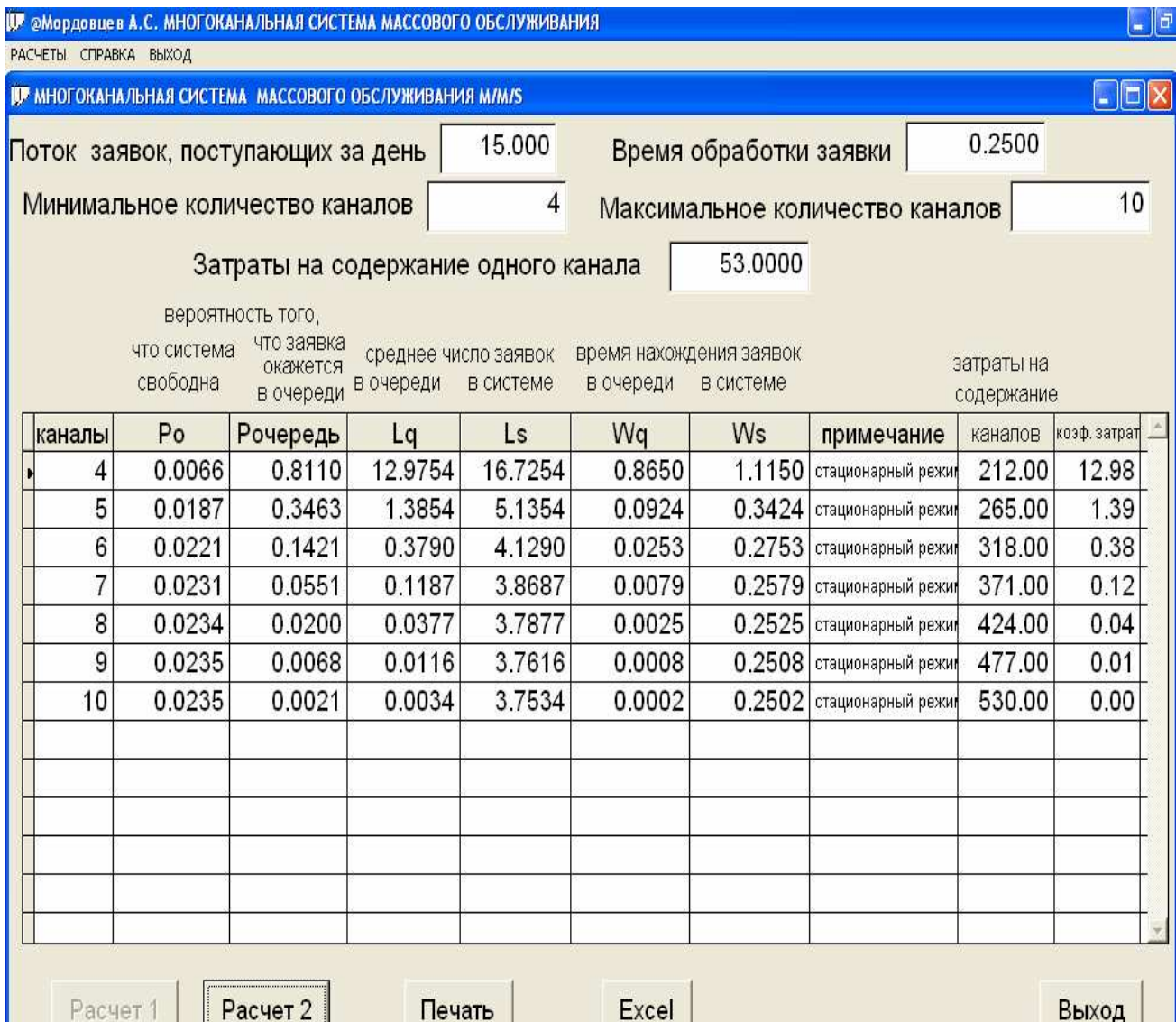


Рис.15.3 Вікно результатів моделювання системи

15.4 Дослідження динаміки обслуговування забудовників

В табл.15.2 наведено дані, що середня тривалість розгляду однієї заявки у відділі чисельністю 6 чоловік дорівнює 3,17 години, або 0,4 дня. Діапазон зміни середньоденного часу розгляду однієї заявки було встановлено в межах від 0,2 до 0,5 дня. Діапазон зміни числа каналів прийнято від 4 до 9. Згідно з табл.15.5 прогнозне середньомісячне число заявок становить $1182/12=99$. Тоді середньоденне число заявок буде $99/22=5$. Перелік витрат на утримання одного каналу наведено в табл. 15.6.

Таблиця 15.6

Зразкові витрати на утримання одного каналу

Витрати	Сума, грн
Зарплата	800
Відрахування	295
Матеріальні витрати	50
Амортизація	15
Інші витрати	40
Всього за місяць	1200
Всього за день	54

Потрібно виявити залежності загальних витрат від тривалості розгляду заявки і від кількості каналів при постійному вхідному потоці, від величини вхідного потоку при фіксованому часі обробки заявки, а також від величини витрат на очікування в черзі для різної кількості каналів.

В табл. 15.7 наведено результати розрахунку характеристик системи для кількості заявок в день – 10 при тривалості обробки однієї заявки - 0,2 дня.

Таблиця 15.7

Результати розрахунку характеристик системи

Кіл-сть каналів	Число заявок (сер)		Тривалість часу		Витрати каналу	Коеф. простою
	В черзі	В системі	В черзі	В системі		
4	0,174	2,174	0,017	0,217	212	0,17
5	0,040	2,040	0,004	0,204	265	0,04
6	0,009	2,009	0,001	0,201	318	0,01
7	0,002	2,002	0	0,20	371	0
8	0	2,0	0	0,2	424	0
9	0	2,0	0	0,2	477	0

В табл. 15.8 і на рис. 15.4 наведено результати розрахунку витрат на обробку заявок для різних значень часу обробки заявки й різного числа каналів при потоці $\lambda = 10$. Сума щоденного збитку від очікування в черзі приймалася рівною 300 грн.

Таблиця 15.8

Витрати на обробку заявок в грн. при $\lambda=10$

Кіл-сть каналів	Час обробки заявок в днях						
	0,2	0,25	0,3	0,35	0,4	0,45	0,5
4	263	371	671	1763			
5	277	304	370	529	931	2323	
6	321	327	348	393	489	696	1200
7	371	374	380	395	425	488	614
8	424	424	427	430	442	463	508
9	477	477	477	480	483	492	507

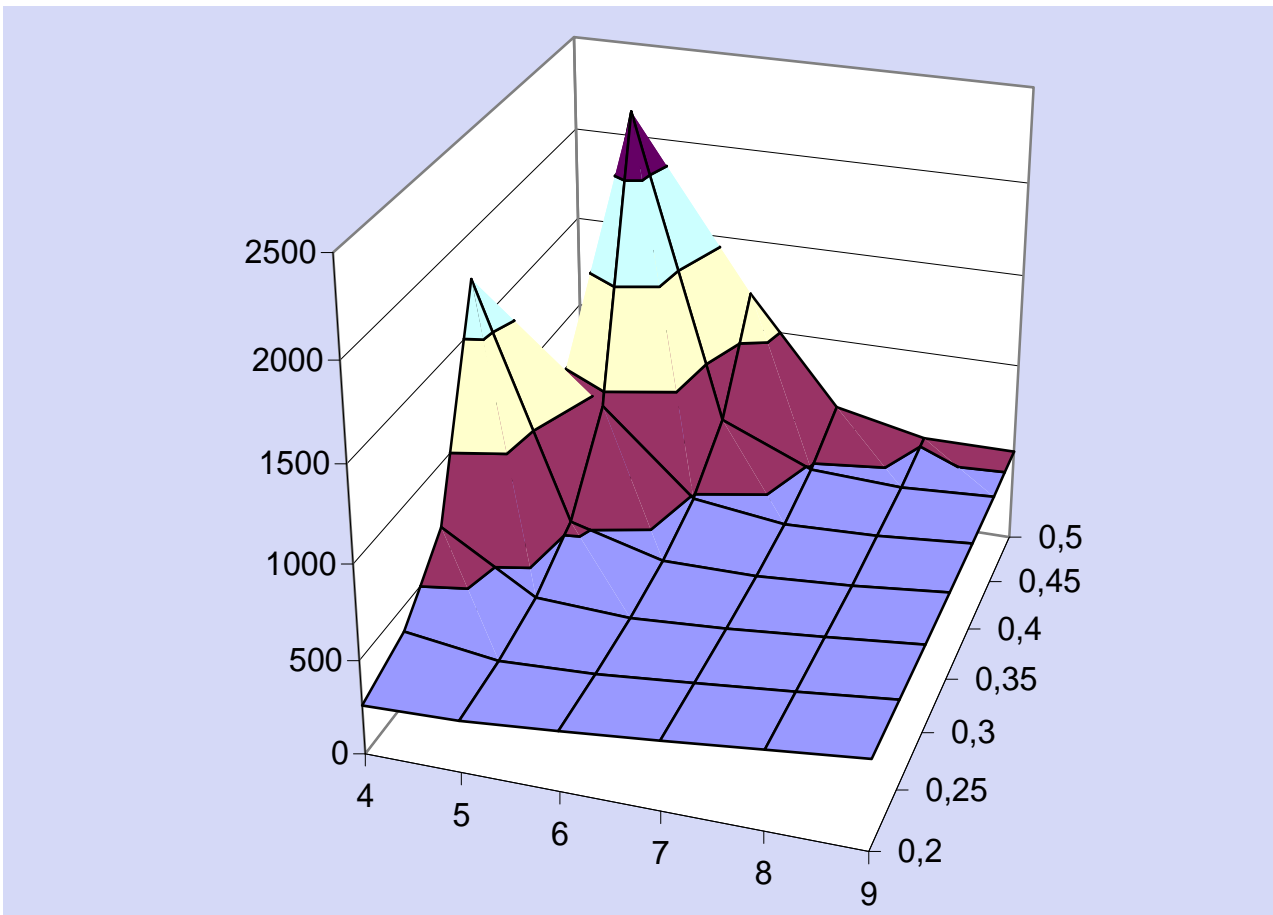


Рис.15.4 Залежність витрат від числа каналів і часу обслуговування

В табл. 15.9 наведено результати розрахунку витрат на обробку заявок, аналогічного попередньому, при потоці $\lambda = 15$. На рис.15.5, 15.6 показані графіки залежності витрат від кількості каналів при $t = 0,25$ (пунктир) і $t = 0,35$ дня (безперервна лінія).

Таблиця 15.9

Витрати на обробку заявок в грн. при $\lambda=15$

Кількість каналів	Час обробки заявок в днях						
	0,2	0,25	0,3	0,35	0,4	0,45	0,5
4	671	4106					
5	370	682	2323				
6	348	432	696	1749			
7	380	407	488	719	1475	7619	
8	427	436	463	541	745	1327	4057
9	477	480	492	519	594	774	1242

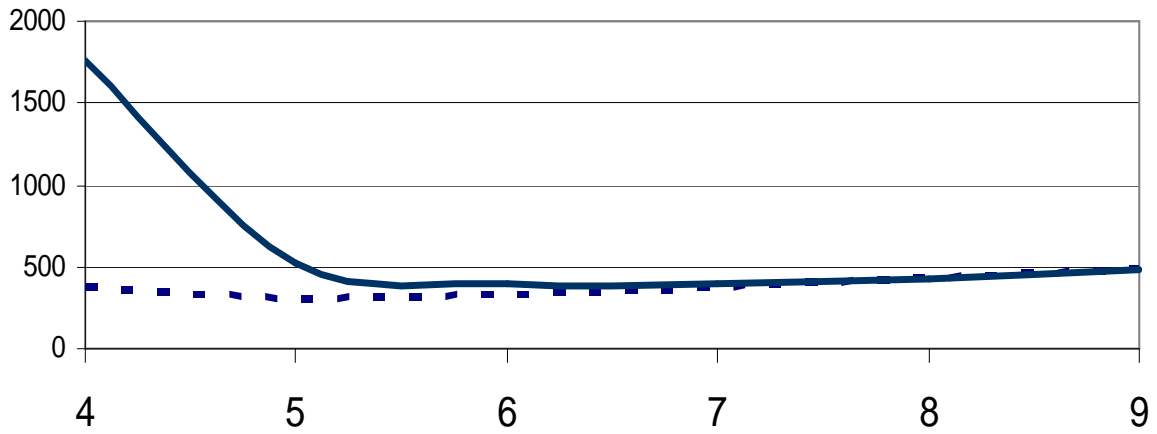


Рис.15.5 Змінення витрат при $\lambda=10$

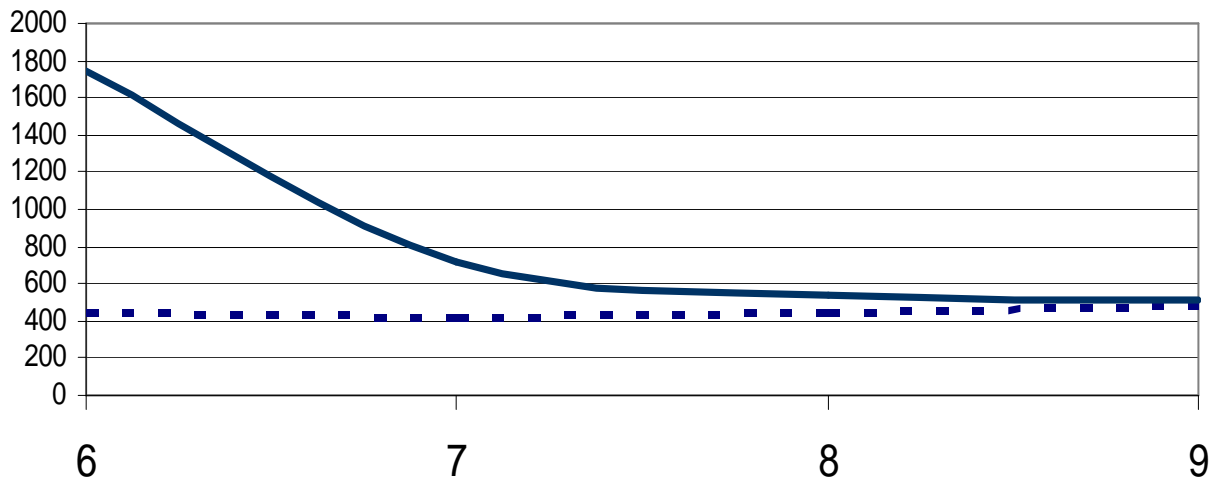


Рис.15.6 Змінення витрат при $\lambda=15$

На **рис.15.7, 15.8** показано графіки залежності витрат від тривалості обробки однієї заявки t для 5 (жирна лінія), 6 (пунктир) і 7 каналів (тонка лінія) при $\lambda =10$ і $\lambda =15$.

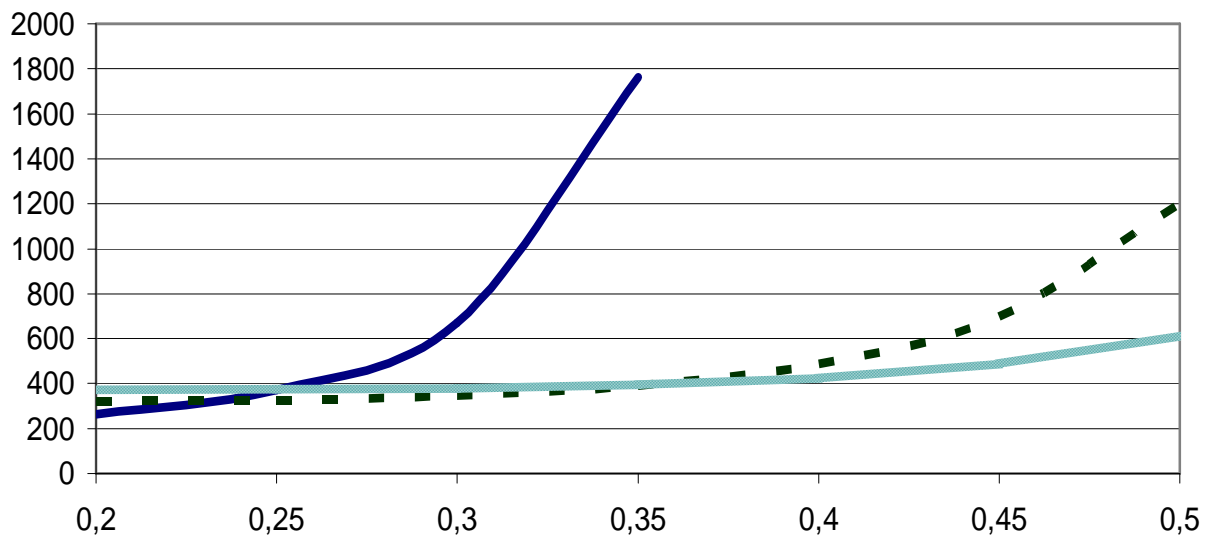


Рис.15.7 Змінення витрат при $\lambda=10$

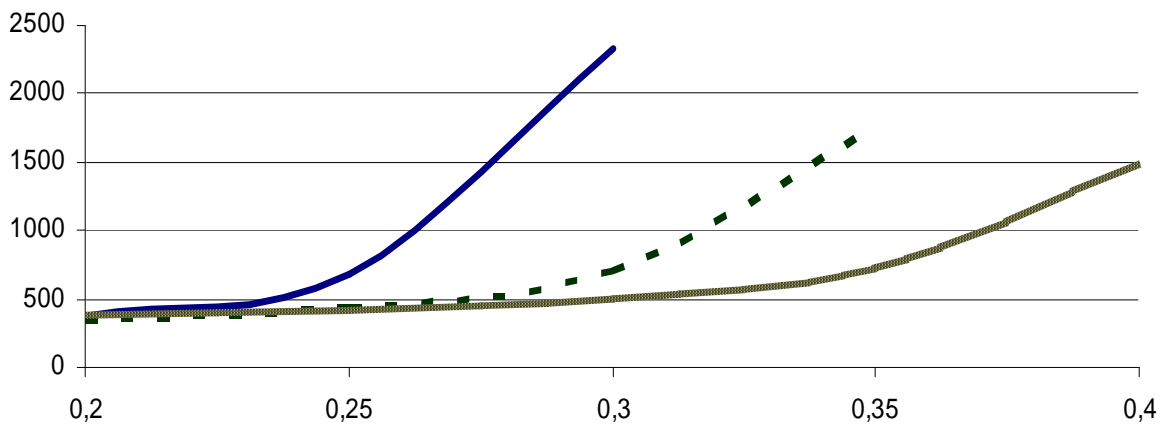


Рис.15.8 Змінення витрат при $\lambda=15$

В табл.15.10 і на рис.15.9 показано залежність витрат від потоку заявок і кількості каналів при $t=0,25$ дня.

Таблиця 15.10

Залежність витрат від потоку заявок і числа каналів

Кількість каналів	Середньоденна кількість заявок						
	8	10	12	14	16	18	20
4	263	371	671	1763			
5	277	304	370	529	931	2323	
6	321	327	348	393	489	696	1200
7	371	374	380	395	425	488	614
8	424	424	427	430	442	463	508
9	477	477	477	480	483	492	507
10	530	530	530	530	533	536	542

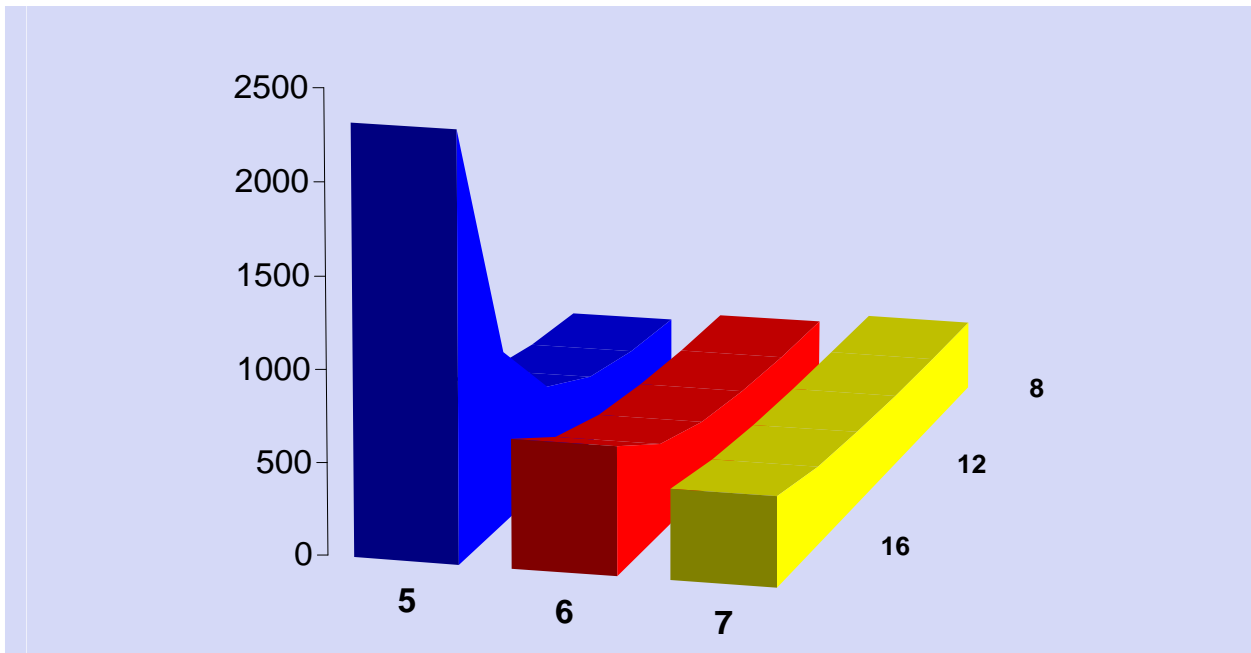


Рис.15.9 Залежність витрат від потоку заявок і числа каналів

На **рис.15.10** показано залежність витрат від кількості каналів при $t = 0,25$ дня і потоці $\lambda = 14$ заявок.

На **рис.15.11, 15.12** показано залежності повних витрат від витрат через очікування в черзі при потоці заявок $\lambda = 10$ для 5-ти (товста безперервна лінія), 6-ти (тонка) і 7-мі (пунктир) каналів обслуговування при $t = 0,25; 0,35$ дня, відповідно.

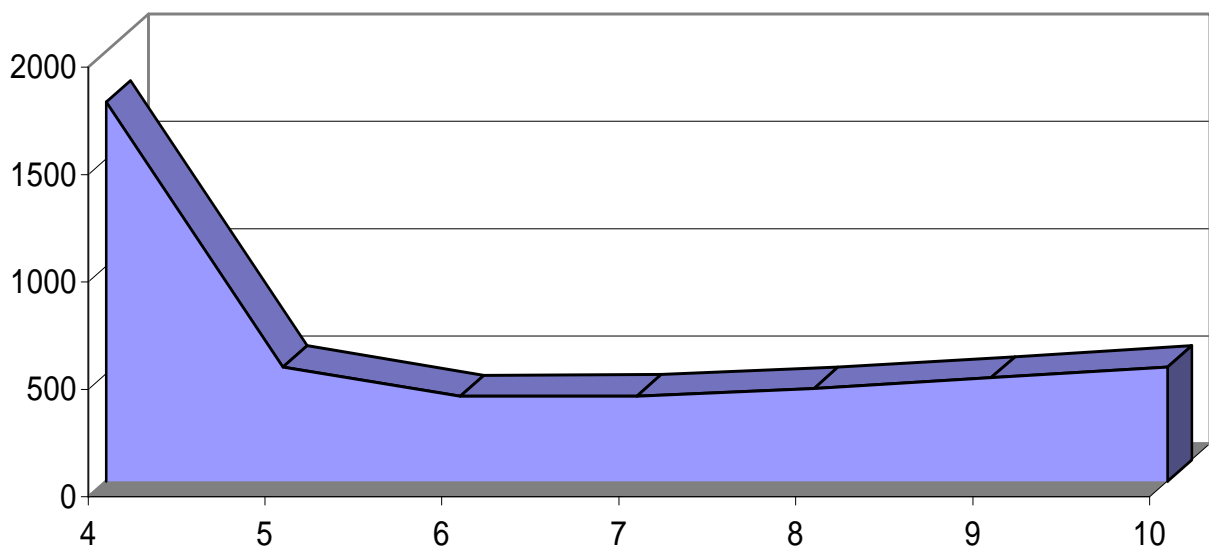


Рис. 15.10 Залежність витрат від числа каналів для потоку з $\lambda = 14$

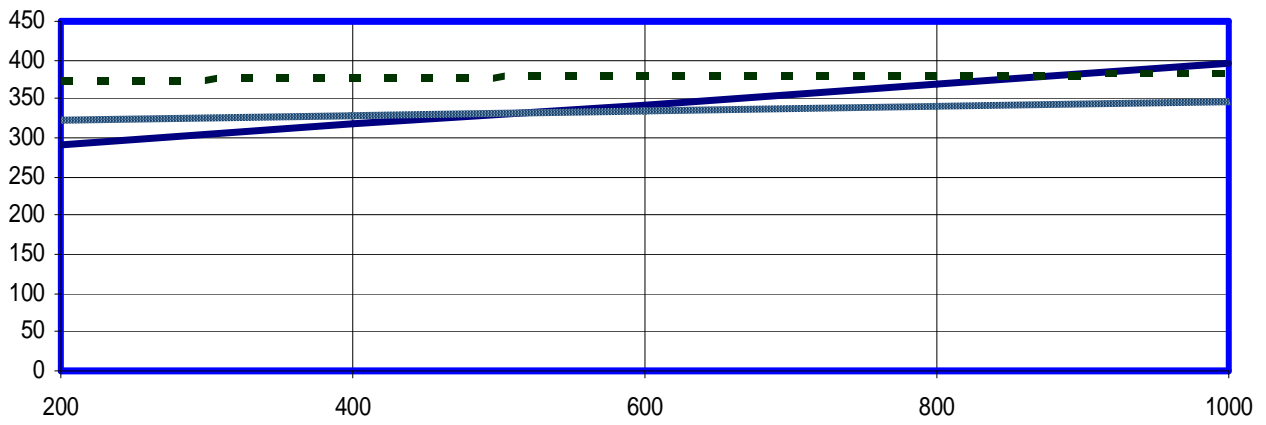


Рис.15.11 Залежність повних витрат від витрат по очікуванню ($t =0,25$ дн)

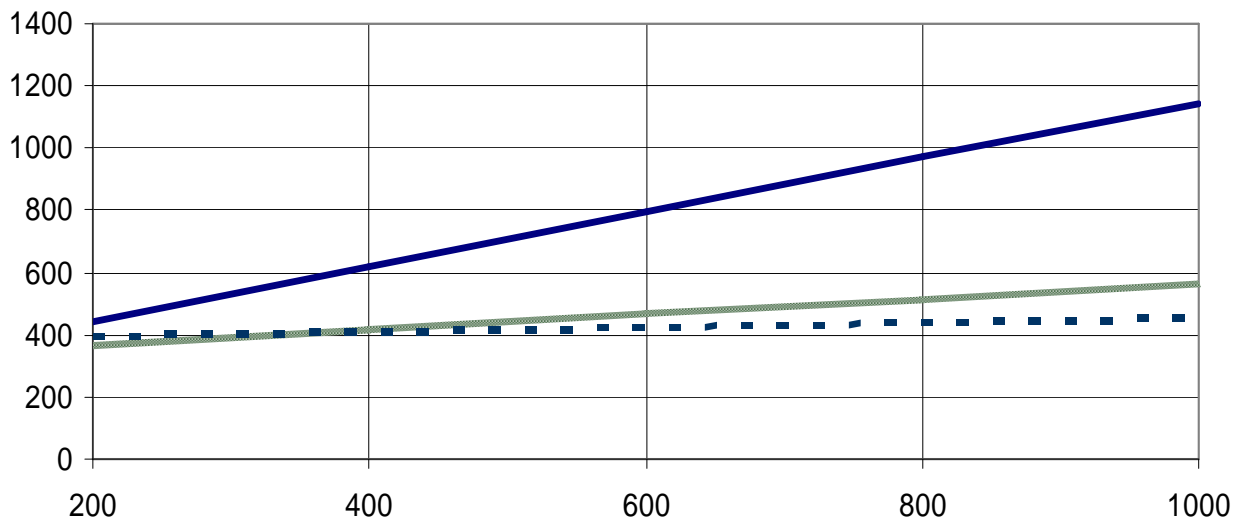


Рис.15.12 Залежність повних витрат від витрат по очікуванню ($t =0,35$ дн)

ВИСНОВКИ

1. При збільшенні потоку заявок збільшується очікування в черзі і, відповідно, втрати від простою заявок: при $\lambda=10$ і середньому часу обробки 0,35 дня щоденні витрати на обробку заявок шістьма співробітниками становлять 393 грн., а при $\lambda=15$ вони зростають до 1749 грн., тобто в 4,5 рази. При збільшенні числа співробітників до 7 чоловік витрати на обробку заявок, при їх потоку $\lambda=15$, стануть 719 грн, тобто зменшуються в 2,4 рази. Це означає, що при такому потоці заявок, додавши одного працівника, можна підвищити ефективність роботи відділу.

2. Впровадження інформаційних технологій для підтримки управлінських рішень дозволить зменшити час обробки заявки. Якщо він буде дорівнювати 0,25 дня, то збільшення потоку заявок з 10 до 15 при 6 співробітниках підвищить витрати тільки в 1,3 рази, а для величини 0,2 дні збільшення потоку практично не відобразитися на витратах. Якщо при цьому зменшити кількість співробітників до 5, то витрати незначно підвищаться.

3. Зменшення часу очікування знижує вплив витрат з очікування в черзі на загальні витрати з оформлення заявок. Наприклад, при $t=0,35$ дня і шести каналах обслуговування повні витрати склали 568 грн. при сумі середньоденних витрат 1000 грн., а при $t=0,25$ дня – 348 грн. В той же час, постійні витрати на обслуговування 6 каналів рівні 318 грн. Тому, в першому випадку сума змінних витрат склала 250 грн., а в другому – 30 грн.

4. Збільшення потоку заявок може привести до того, що пропускна здатність ряду структур міського виконкому буде недостатньою. Заявки будуть довго чекати своєї черги. Тому необхідно при зміні потоку забезпечити можливість гнучко перебудовувати процес, щоб зменшити час

очікування, і на кожному етапі процесу треба контролювати й регулювати очікування. Надалі необхідно провести такі дослідження для кожного відділу і за їх результатами прийняти відповідні рішення щодо підвищення ефективності їх роботи.

5. При збільшенні потоку вхідних заявок з 8 до 18 (**табл.15.10** і **рис.15.9, 15.10**) витрати на їх обробку підвищуються таким чином: для 5 каналів – в 8,4 рази, для 6 каналів – в 2,2 рази, для 7 каналів – в 1,3 рази. Це означає, що скорочення чисельності співробітників відділу при такому рості потоку заявок нераціонально, а додавання співробітників дозволить зменшити витрати. Із **рис.15.10** також видно, що при фіксованому потоці заявок $\lambda=14$ збільшення числа каналів спочатку зменшує витрати до мінімального значення, а потім вони знову зростають.

СПИСОК ЛИТЕРАТУРЫ

1. Адлер Ю.П. и др. *Планирование экспериментов при поиске оптимальных условий*. – М.: Наука, 1971. – 215 с.
2. Бахвалов Л.А. *Компьютерное моделирование: долгий путь к сияющим вершинам*// Компьютер. – 1997. – № 40. – С.26-36.
3. Бусленко Н.П., Калашников В.В и др. *Лекции по теории сложных систем*. – М.: Сов. радио, 1973. - 440 с.
4. Бусленко Н.П. *Моделирование сложных систем*. – М.: Наука, 1978. – 400 с.
5. Вавилов А.А. *Имитационное моделирование производственных систем*. – М.: Машиностроение; Берлин: Техника, 1983. – 416 с.
6. Варжапетян А.Г. *Имитационное моделирование на GPSS/H*. – М.: Вузовская книга, 2004. – 256 с.
7. Горбунов А. Р. *Пакет моделирования ITHINK: инвестиционные проекты, реинжиниринг, стратегия*. – М.: ТОРА-Центр, 1997. – 24 с.
8. Емельянов А.А., Власова Е.А. *Имитационное моделирование в экономических информационных системах*. – М.: МЭСИ, 1998. – 108 с.
9. Емельянов А.А. *Имитационное моделирование в управлении рисками*. – СПб.: Инжекон, 2000. – 376 с.
10. Емельянов А.А., Власова Е.А., Дума Р.В. *Имитационное моделирование экономических процессов*. – М.: Финансы и статистика, 2004. – 368 с.
11. *Имитационные системы принятия экономических решений* / К.А Багриновский и др. – М.: Наука, 1989. – 185 с.
12. *Имитационный анализ регионального воспроизводства*. /Отв.ред. В.Г.Булавский. – Новосибирск: Наука СО, 1987. – 176 с.
13. Калашников В.В. *Организация моделирования сложных систем*. – М.: Знание, 1982. – 62 с.
14. Киндлер Е. *Языки моделирования: Пер. с чеш.* – М.: Энергоатомиздат, 1985. – 288 с.
15. Клейнен Дж. *Статистические методы в имитационном моделировании*. В 2-х т. – М.: Статистика, 1978. – 462 с.
16. Клир Дж. *Системология. Автоматизация решения системных задач*. – М.: Радио и связь, 1990. – 293 с.
17. Коблев Н.Б. *Основы имитационного моделирования сложных экономических систем*. – М.: Дело, 2003. – 320 с.
18. Кугаенко А.А. *Основы теории и практики динамического моделирования социально-экономических объектов и прогнозирования их развития*. – М.: Вузовская книга 1998. – 392 с.

19. Кузнецов Ю.А., Перова В.И., Мичасова О.В. *Работа с программным пакетом ITNINK*: учебное пособие. - Нижний Новгород: НГГУ, 2005. – 72 с.
20. Кузнецов Ю.А., Перова В.И., Мичасова О.В. *Имитационное моделирование экономических процессов с применением программного пакета ITNINK // Экономический анализ: теория и практика.* – 2006. – №6. – С.11-15.
21. Кузнецов Ю.А., Перова В.И. *Применение пакета имитационного моделирования для анализа математических моделей экономических систем.* – Нижний Новгород: НГГУ, 2007. – 99 с.
22. Лоу А.М., Кельтон В.Д. *Имитационное моделирование. Классика CS.* 3-е изд. – М.: Питер, 2004. – 421 с.
23. Лычкина Н.Н. *Современные тенденции в имитационном моделировании // Вестник университета, серия “ИС управления”.* – № 2. – М.: ГУУ, 2000.
24. Лычкина Н.Н. *Технологические возможности современных систем моделирования // Банковские технологии. Вып. 9.* – М., 2000.
25. Лычкина Н.Н. *Компьютерное моделирование социально-экономического развития регионов в системах поддержки принятия решений // III Межд конф “Идентификация систем и задачи управления”.* – М.: SICPRO’04, 2004.
26. Лычкина Н.Н. *Имитационное моделирование экономических процессов: учеб. пособие.* – М.: Академия АйТи, 2005. – 163 с.
27. Максимей И.В. *Имитационное моделирование на ЭВМ.* – М.: Радио и связь, 1988.- 232с.
28. Нейлор Т. *Машинные имитационные эксперименты с моделями экономических систем:* пер. с англ. – М.: Мир, 1975. – 502 с.
29. Панасенко А.А., Намяк Д.Е. *Информационно-аналитические системы предприятий ВКХ.* – Харьков: Основа, 2005. – 192 с.
29. Перегудов Ф.И. Тарасенко Ф.П. *Введение в системный анализ: уч пособие.* – М.: Высшая школа, 1989. – 367 с.
30. Прангишвили И.В. *Системный подход и общесистемные закономерности.* – М.: СИНТЕГ, 2000. – 528 с.
31. Прицкер А. *Введение в имитационное моделирование и язык СЛАМ II:* пер. с англ. – М: Мир, 1987. – 646 с.
32. Ресин В.И., Попков Ю.С. *Развитие больших городов в условиях переходной экономики. Системный подход.* – М.: Эдиториал УРСС, 2000.
33. *Руководство пользователя по GPSS World:* пер. с англ. – Казань: Мастер Лайн, 2002. – 384 с.
34. Рыжиков Д.И. *Имитационное моделирование систем массового обслуживания.* – Л.: ВИККИ им А.Ф. Можайского, 1991. – 111 с.
35. Сидоренко В.Н. *Системная динамика.* - М.: ТЕИС, 1998. – 205 с.

36. Смирнов В.С., Власов С.А. и др. *Методы и модели управления проектами в металлургии*. – М.: СИНТЕГ, 2001. – 176 с.
37. Советов Б.Я., Яковлев С.А. *Моделирование систем: учебник*. – М.: Высшая школа, 1985. – 224 с.
38. Советов Б.Я., Яковлев С.А. *Моделирование систем*. – М.: ВШ, 2003. – 320 с.
39. *Социальные системы. Формализация и компьютерное моделирование: учебное пособие* /А.К.Гуц, В.В.Коробицын и др. – Омск: ОГУ, 2000. – 160 с.
40. *Технология системного моделирования*/Под общ.ред. С.В. Емельянова. – М.: Машиностроение, 1998.
41. Томашевский В., Жданова Е. *Имитационное моделирование в среде GPSS*. – М: Бестселлер, 2003. – 416 с.
42. Трахтенгерц Э.А. *Компьютерная поддержка принятия решений: Научно-практическое издание. Серия “Информатизация России на пороге XXI века”*. – М.: СИНТЕГ, 1998.
43. Трахтенгерц Э.А. *Субъективность в компьютерной поддержке управленческих решений*. – М.: СИНТЕГ, 2001. – 256 с.
44. Форрестер Дж. *Основы кибернетики предприятия (индустриальная динамика)*: пер. с англ. – М.: Прогресс, 1971. – 340 с.
45. Форрестер Дж. *Динамика развития города*. – М.: Прогресс, 1974.
46. Форрестер Дж. *Мировая динамика*. – М.: Наука, 1978.
47. Цвиркун А.Д., Акинфиев В.К.и др. *Имитационное моделирование в задачах синтеза структуры сложных систем (оптимизационно-имитационный подход)*. – М.: Наука, 1985. – 176 с.
48. Шебеко Ю.А. *ITHINK - финансовым менеджерам (аналитический пакет нового поколения помогает в решении проблем)* // www.tora-centre.ru
49. Шеннон Р. *Имитационное моделирование систем – искусство и наука*. – М.: Мир, 1978.
50. Шрайбер Т. Дж. *Моделирование на GPSS*. – М.: Машиностроение, 1980. – 592 с.
51. Шутенко Л.Н., Торкатюк В.И., Пан Н.П.и др. *Моделирование восстановления инфраструктуры ВКХ Харькова при реализации энергосберегающих проектов* // Энергетика. Энергоаудит. – Харьков, 2006. – №7. – С.29-38.
52. Яцкив И.В. *Проблема валидации имитационной модели и ее возможные решения*. Материалы конф. ИММОД. – 2003. – С. 211-217.
53. Balci O. *Credibility Assessment of Simulation Results*//Proceedings of the 1986 Winter Simulation Conference. – 1986. – pp. 39-44.
54. Balci O. *Validation, Verification and Testing Techniques Throughout the Life Cycle of a Simulation Study*, Annals of Operation Research. – 1994.

55. Balci O. *Verification, validation and accreditation*//Proceedings of the 1998 Winter Simulation Conference. – 1998. – pp. 41-48.
56. Bass F.M. *A new product growth for consumer durables* // Management science. 1969. Vol.15.
57. Carson J.S. *Model verification and validation* // Proceedings of the 2002 Winter Simulation Conference. – 2002. – pp. 52-58.
58. Fareena Sultan, Jonn U.Farley, Donald R.Lehmann. *A Meta- analysis of Diffusion Models* // Journal of Marketing Research. – 1990.
59. Henriksen J. *An Introduction to SLX* //Proceedings of the 1997 Winter Simulation Conference IEEE. – 1997. – pp. 593-599.
60. Kelton W., Randall P., Sadowski, Deborah A. *Simulation with Arena – WCB / McGraw-Hill*. – 1998.
61. Law A.M., McComas M.G. *How to build valid and credible simulation models*// Proceedings of the 2001 Winter Simulation Conference. – 2001. – pp. 22-29.
62. Sargent R.G. *Some approaches and paradigms for verifying and validating simulation models* // Proceedings of the 2001 Winter Simulation Conference. – 2001. – pp. 106-114.
63. Sterman J.D. *System Dynamics Models for Project Management*. www.rub.ruc.dk
64. www.economy.mari.ru Рынок программных средств.

НАВЧАЛЬНЕ ВИДАННЯ

**УДОСКОНАЛЕННЯ
БІЗНЕС-СИСТЕМ**

Навчальний посібник

Том 2

**Аналіз і удосконалення бізнес-систем
динамічними методами**

Автори: Володимир Олександрович Лелюк,
Олексій Володимирович Лелюк,
Микола Павлович Пан.

Відповідальний за випуск: Є. М. Кайлюк

Редактор: М. З. Аляб'єв

Верстка: Ю.П. Степась

План 2009, поз. 33 Н

Підп. до друку 12.01.2010	Формат 60X84 1/16	Папір офісний
Друк на ризографі	Умовн.- друк. арк. 5,1	Обл.- вид. арк. 5,5
Зам. №	Тираж 500 прим.	

Харківська національна академія міського господарства,
61002, Харків, вул. Революції, 12

Сектор оперативної поліграфії ЦНІТ ХНАМГ
ХНАМГ, 61002, Харків, вул. Революції, 12