

## РЕАЛІЗАЦІЯ ТИРАЖУВАННЯ НЕЗАЛЕЖНИХ СЕГМЕНТІВ ОБЧИСЛЮВАЛЬНОГО ЕКСПЕРИМЕНТУ В РОЗПОДІЛЕНІЙ ДИСКРЕТНО-ПОДІЙНІЙ СИСТЕМІ ІМІТАЦІЙНОГО МОДЕЛЮВАННЯ OPENGPSS

### Вступ

Імітаційне моделювання є одним з важливих методів аналізу різних складних систем. Разом зі збільшенням продуктивності та складності сучасних комп'ютерних систем постають нові вимоги до систем імітаційного моделювання – прискорення отримання результатів моделювання. Для цього можна використовувати розподілені системи імітаційного моделювання, які розглянуті в [1].

Сьогодні існує багато реалізацій розподілених систем та систем тиражування, наприклад, SPEEDES [2] або PARASOL [3], і розробники, які працюють із поширеною мовою імітаційного моделювання GPSS [4], вважають за необхідним підвищення швидкості моделювання своїх моделей. Тому виникає проблема побудови системи моделювання з автоматичним розділенням GPSS-програм.

### Постановка задачі

Цілі дослідження: необхідно побудувати алгоритм розділення завдання моделювання на незалежні частини, реалізувати алгоритм розділення, роздати завдання вузлам кластера та збирання результатів, дослідити роботу цього алгоритму.

### Сегменти обчислювального експерименту

Розглянемо *обчислювальний експеримент (ОЕ)* як етап імітаційного моделювання, під час якого проводяться прогони вже побудованої імітаційної моделі на комп'ютері при різних вхідних параметрах.

Розподілена дискретно-подійна система імітаційного моделювання OpenGPSS, яка доступна в режимі онлайн за адресою [5], використовує мову імітаційного моделювання GPSS, в якій є команди керування ОЕ. Експеримент може бути факторним, оптимізаційним або аналізом чутливості системи. Тип експерименту задає сам користувач за допомогою послідовності команд керування обчислювальним експериментом.

Розглянемо типовий обчислювальний експеримент мовою GPSS, який складається з декількох прогонів (табл. 1).

Експеримент складається з викликів команд керування RMULT, START, RESULT, CLEAR, ANOVA.

Таблиця 1. Уривок тексту GPSS-програми.

№	Команди GPSS
	...
110	RMULT 1, 2, 8, 5, 8, 2, 8, 1
120	START 1

130	RESULT 2,V\$WORKER
140	CLEAR
150	RMULT 7, 4, 1, 12, 4, 3, 0, 2
160	START 1
170	RESULT 2,V\$WORKER
200	ANOVA

За допомогою команди RMULT встановлюються початкові стани множників для всіх датчиків псевдовипадкових чисел (ДПЧ). Команда START запускає ОЕ. У GPSS-програмі можна кілька разів використовувати команду START для отримання проміжних результатів моделювання. Команда CLEAR знищує всю статистику системи та всі транзакти. Після виконання команди CLEAR уся система, крім ДПЧ, перебуває в початковому стані. Команда RESULT заносить результати моделювання в тимчасовий буфер результатів для збереження значень і наступного аналізу. Команда ANOVA виконує дисперсійний аналіз отриманих результатів.

Введемо визначення, які необхідні для подальшого розгляду матеріалу статті.

Нехай *сегмент обчислювального експерименту* (COE) - це послідовність команд керування, яка починається командою CLEAR або RMULT, містить не більше однієї команди CLEAR, RMULT, START та закінчується командою START або RESULT. Сегменти обчислювального експерименту розділяються на два типи: залежні і незалежні.

Будемо називати *залежним сегментом обчислювального експерименту* (ЗСОЕ) такі COE, які не містять команду RMULT або не містять команду CLEAR. Дані в таких сегментах залежать від передісторії станів системи.

*Незалежні сегменти обчислювального експерименту* (НСОЕ) – це такі COE, які обов'язково складаються з послідовності команд CLEAR, RMULT, START. Саме послідовність команд CLEAR, RMULT заставляє систему “забути” свої попередні стани, тобто сегмент не залежить від передісторії станів системи.

Будемо називати *кадром сегмента обчислювального експерименту* (КСОЕ) – послідовність COE, яка починається з одного НСОЕ, та включає в себе всі ЗСОЕ, які залежать від цього початкового сегмента.

Розділимо GPSS-програму з табл. 1 на сегменти, отримаємо два НСОЕ, які наведені в табл. 2. COE 1 – незалежний, тому що це перший COE та його виконання ні від чого не залежить; COE 2 – також незалежний, тому що виконуються команди CLEAR, RMULT, які встановлюють “початковий” стан системи.

Таблиця 2. Уривок тексту GPSS-програми з сегментами.

№	Команди GPSS	Сегменти
	...	
110	RMULT 1, 2, 8, 5, 8, 2, 8, 1	COE1
120	START 1	

130	RESULT 2,V\$WORKER	
140	CLEAR	COE2
150	RMULT 7, 4, 1, 12, 4, 3, 0, 2	
160	START 1	
170	RESULT 2,V\$WORKER	
200	ANOVA	Аналіз результатів

З прикладу видно, що початкові значення всіх ДПЧ, які використовувались раніше обов'язково повинні бути перевизначені командою RMULT. Команда ANOVA застосовується для дисперсійного аналізу після етапу збирання результатів.

### Тиражування

Система OpenGPSS має агентну архітектуру, яка описана в [6]. Нехай  $A_i = \{A_i^{AgSm}, A_i^{AgRep}, A_i^{AgUst}\}$  – множина активних агентів  $i$ -го вузла,  $i = \overline{1, n}$ , де  $n$  – кількість вузлів у кластері моделювання.

Введемо матрицю  $\mathbf{V}_{iz}$  розмірності  $n \times n$ , котра встановлює зв'язок між  $i$ -м і  $z$ -м вузлом:

$$\mathbf{V}_{iz} = \begin{cases} 1, \text{ якщо існує зв'язок між } i\text{-им та } z\text{-им вузлом;} \\ 0, \text{ у протилежному випадку;} \end{cases}, \quad i = \overline{1, n}, \quad z = \overline{1, n}.$$

Припустимо, що  $S_{ij}$  –  $j$ -а імітаційна модель, яка виконується на  $i$ -му вузлі,  $i = \overline{1, n}$ ,  $j = \overline{1, n_i^j}$ , де  $n_i^j$  – кількість моделей на  $i$ -му вузлі. У кожній моделі  $i$ -го вузла є свій модельний час, множину яких позначимо  $T_i = \{t_{i1}, t_{i2}, \dots, t_{in_i^j}\}$ .

Наявність послідовності команд CLEAR, RMULT, START, дозволяє виділити незалежні СОЕ, множину яких позначимо  $U_{ij} = \{u_{ij1}, u_{ij2}, \dots, u_{ijn_i^j}\}$ . Всі інші СОЕ відносимо до залежних, множина яких –  $C_{ij} = \{c_{ij1}, c_{ij2}, \dots, c_{ijn_i^j}\}$ . Зауважимо, що перший СОЕ в моделі завжди незалежний, тому він належить множині  $U_{ij} = \{u_{ij1}, u_{ij2}, \dots, u_{ijn_i^j}\}$ .

Побудуємо кадр сегмента обчислювального експерименту як послідовність СОЕ, що починається з одного НСОЕ, та включає в себе всі ЗСОЕ, які залежать від цього початкового сегмента:

$$K_{ijm} = \{u_{ijm} \mid \forall m: u_{ijm} \in U_{ij}\} \cup \{c_{ifr} \mid c_{ifr} \in C_{ij}, c_{ifr} \text{ залежить від } u_{ijm}\}, \quad i = \overline{1, n}, \quad j = \overline{1, n_i^j}, \quad m = \overline{1, n_i^u}.$$

З побудови КСОЕ впливає незалежність кадрів один від одного.

Нехай  $K_{ij}$  – множина КСОЕ на  $i$ -му вузлі для  $j$ -ої моделі:

$$K_i = \bigcup_{n_i^j}^{j=1} K_{ij},$$

а результати проведення всіх СОЕ з КСОЕ позначимо  $R_{ij}$ :

$$R_j = \bigcup_{n_j^k}^{m-1} R_{ijm}.$$

Об'єднання результатів по всім КСОЕ  $j$ -ої моделі дає загальний результат:

$$R_i = \bigcup_{n_i^j}^{j-1} R_{ij}.$$

Тобто проведення одного КСОЕ дає один і тільки один результат з множини  $R_i$ .

Стан  $i$ -го вузла опишемо як  $P_i(A, S_i, T_i, K_i, R_i)$ , а стан кластера моделювання –  $P = \bigcup_n^{i-1} P_i(A, S_i, T_i, K_i, R_i)$ .

Незалежність КСОЕ дозволяє проводити *тиражування* – одночасне виконання КСОЕ на різних вузлах кластера імітаційного моделювання, з подальшим об'єднанням результатів.

Система OpenGPSS автоматично виділяє множини КСОЕ, які можуть одночасно виконуватись на різних вузлах кластера. Після запуску  $j$ -ої GPSS-моделі агент користувача  $A_i^{AgUsr}$ , який виконує функцію компілятора [7] переводить GPSS-текст у внутрішній формат та виділяє кадри СОЕ:

$$A_i^{AgUsr} : P_i(S_i, K_i) \rightarrow P_i(S_i \cup \{s_{ij}\}, K_i \cup \{k_{ij}\}).$$

Агент реплікації AgRep [8] копіює GPSS-модель та всі КСОЕ на інші вузли:

$$A_i^{AgRep} : P_i(S_i, K_i) \rightarrow P_z(S_z \cup S_i, K_z \cup K_i), \forall i, z : \mathbf{B}_{iz} = 1,$$

Тому кожний вузол повністю зберігає інформацію про модель та експеримент для запобігання втрат даних у разі вимкнення одного з вузлів.

У кластері немає централізованого керування запуском КСОЕ – кожний вузол випадково визначає КСОЕ із множини ще необроблених сегментів. Тим самим уникаються конфлікти блокування: жоден вузол не чекає на результати обробки ОЕ іншими вузлами – вони діють за оптимістичним планом – продовжують виконувати моделювання над сегментами. Розглянутий підхід, на відміну від підхода з динамічним розподілом на підмножини вузлів кластера та визначення локального вузла-координатора за допомогою одного з алгоритмів голосування, які наведені в [9], має більшу масштабованість (канал зв'язку координатора може бути перевантаженим) та високу надійність (при відмові координатора необхідно чекати переголосування).

Агент імітаційного моделювання AgSim виконує КСОЕ на  $i$ -му вузлі:

$$A_i^{AgSim} : P_i(\{t_{i1}, t_{i2}, \dots, t_{ij}, \dots, t_{in_i}\}) \rightarrow P_i(\{t_{i1}, t_{i2}, \dots, t_{ij}, \dots, t_{in_i}\}), \text{ де } t_{ij} \leq t_{ij}.$$

### Збирання результатів

Агент реплікації AgRep, реалізований мовою PL/SQL [10, 11], періодично копіює таблиці результатів REPORT и REPORT\_DETAIL на інші вузли:

$$A_i^{AgRep} : P_z(R_z) \rightarrow P_i(R_i \cup R_z), \forall i, z : \mathbf{B}_{iz} = 1.$$

Потім AgRep виконує *збирання результатів* – об'єднання результатів всіх СОЕ для подальшого виконання і аналізу загальних результатів (наприклад, виконання команди ANOVA) та завершує ОЕ.

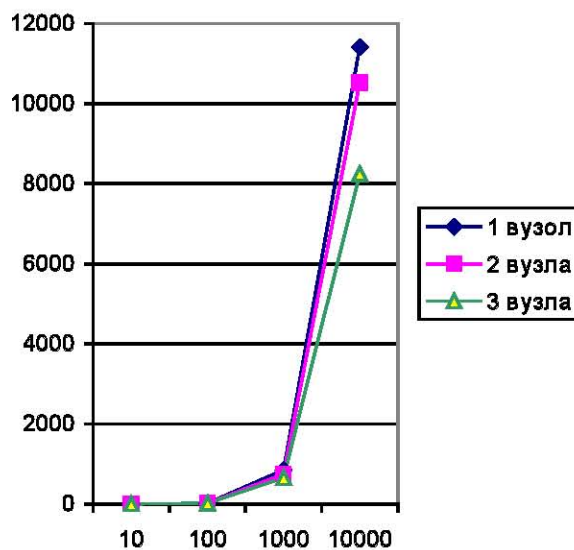
### Отримані результати

Проведемо з GPSS-програмою, яка наведена в табл. 3, серію експериментів при різній кількості вузлів кластера і різному значенні кількості транзактів – будемо змінювати лічильник завершень у команді START.

Таблиця 3. Текст тестової GPSS-програми.

№	Команди GPSS
10	GENERATE 5
20	QUEUE QUE1
30	SEIZE PRIB1
40	DEPART QUE1
50	ADVANCE 2
60	RELEASE PRIB1
70	TERMINATE 1
80	RMULT 1, 2, 8, 5, 8, 2, 8, 1
90	START 10
100	CLEAR
110	RMULT 1, 2, 8, 5, 8, 2, 8, 1
120	START 10
130	CLEAR
140	RMULT 7, 4, 1, 12, 4, 3, 0, 2
150	START 10
200	ANOVA

Результати проведення експериментів відображені на рисунку. По осі абсцис відкладені значення лічильника завершень у команді START, по осі ординат астрономічний час в секундах.



Залежність часу моделювання від кількості вузлів кластера.

## Висновки

1. Запропонований підхід розділення експерименту на незалежні частини для подальшого паралельного виконання на вузлах кластера та збирання результатів призводить до підвищення продуктивності системи – зменшення часу моделювання.

2. Застосований агентний підхід для реалізації системи моделювання підвищує масштабованість, відмовостійкість та надійність кластера при незначному зниженні продуктивності та надмірності даних для підтримання цілісності моделі та експерименту.

3. Отримані результати підтверджують можливість автоматичного розділення експерименту на незалежні частини для прискорення швидкості моделювання.

В подальшому можна досліджувати оптимізацію режимів роботи кластера для підвищення швидкості моделювання, будувати та досліджувати кластери з великою кількістю вузлів (10-20), розподілювати дискретно-подійну систему імітаційного моделювання з оптимістичною синхронізацією модельного часу у вузлах кластера.

### Література

1. *Richard M. Fujimoto*. Parallel And Distributed Simulation Systems. Wiley, 2000.
2. *SPEEDES*. <http://www.speedes.com>.
3. *Mascarenhas E., Knop F., Vernon R.* ParaSol: A multithreaded system for parallel simulation based on mobile threads. Winter Simulation Conference, 1995.
4. *Шрайбер Томас Дж.* Моделирование с использованием GPSS. – М.: Машиностроение, 1980.
5. *Киевский Центр Имитационного Моделирования*. <http://www.simulation.kiev.ua>.
6. *Томашевский В.Н., Диденко Д.Г.* Агентная архитектура распределенной дискретно-событийной системы имитационного моделирования OpenGPSS. Системні дослідження та інформаційні технології. № 4, 2006. – К.: ВПК “Політехніка”, 2006. – с.123–133.
7. *Ахо Альфред, Сети Рави, Ульман Джеффри.* Компиляторы: принципы, технологии, инструменты. Пер. с англ. – М.: Издательский дом "Вильямс", 2001. – 768 с.
8. *Диденко Д.Г.* Агент реплікації в розподіленій дискретно-подійній системі імітаційного моделювання OPENGPSS. Матеріали міжнародної наукової конференції “Інтелектуальні системи прийняття рішень та прикладні аспекти інформаційних технологій”, 2006. – с. 264–266.
9. *Таненбаум Э., Стеен ван М.* Распределенные системы. Принципы и парадигмы. – СПб.: Питер, 2003. – 877 с.
10. *Кайт Том.* Oracle для профессионалов. Книга 2. Расширение возможностей и защита: Пер. с англ. – М.: ДиасофтЮП, 2003. – 848 с.
11. *Кайт Том.* Oracle для профессионалов. Книга 1. Архитектура и основные особенности: Пер. с англ. – М.: ДиасофтЮП, 2003. – 672 с.