

*Sublanguages/Codd E/F//Database systems. – 1972, vol 54, #2.-P.65-98. 3. Плоткин Б.И. Универсальная алгебра, алгебраическая логика и базы данных / Б.И. Плоткин. – М.: Наука, 1991. – 446 с. 4. Бениаминов Е.М. Алгебраические методы в теории без данных и представлений знаний / Бениаминов Е.М. – М.: Научный мир, 2003. – С.184. 5. Koo B Algebra of systems: a metalanguage for model synthesis and evaluation / Koo B, Simmons W. // IEEE Transactions on systems, man and cybernetics, vol 39, N 3, may 2009 6. Буров С.В. Концептуальне моделювання інтелектуальних програмних систем: монографія / С.В. Буров. – Львів: Вид-во Львівської політехніки, 2012. – 432 с. 7. Mitsuri Ikeda. Task ontology: Ontology for building conceptual problem solving models/ Mitsuri Ikeda, Kazuhisa Seta, Osamu Kakusho, Riichiro Mizoguchi. // Workshop note of application of ontologies and problem-solving methods, ECAI98, 1998 8. Johnson, Peter. Task-Related Knowledge Structures: Analysis, Modelling and Application / Johnson Peter, Johnson Hilary, Waddington Ray, Shouls, Alan// Knowledge Creation Diffusion Utilization, 1988. – P. 35–62 9. Van Welie Martin. An Ontology for Task World Models / Van Welie, Martijn, Van Der Veer, Gerrit C Eliëns // Methods, v. 98, 1998. – P. 57–70. 10. Raubal Martin. Ontology-Based Task Simulation/ Raubal M., Kuhn W. // Spatial Cognition and Computation, v.4, 2004. – P. 15–37 11. Seta Kazuhisa. Building ontologies for conceptual model management / Seta Kazuhisa, Koyama Kazuya, Hayashi Yusuke, Ikeda Mitsuru // WSEAS Transactions on Information Science and Applications, v.3, 2006. – P. 546–553. 12. Erl, T. SOA Principles of Service Design. / Erl, T. – Prentice Hall, 2007.*

УДК 004.652

О.М. Верес<sup>1</sup>, Ю.О. Верес<sup>2</sup>

Національний університет “Львівська політехніка”,

<sup>1</sup>кафедра інформаційних систем та мереж;

<sup>2</sup>кафедра соціальних комунікацій та інформаційної діяльності

## ЗАСТОСУВАННЯ ОБ’ЄКТНО-ОРІЄНТОВАНОГО ПІДХОДУ ДО ПОБУДОВИ МОДЕЛІ СППР

© Верес О.М., Верес Ю.О., 2013

Описано об’єктно-орієнтований підхід моделювання СППР. Запропоновано та описано трирівневу архітектуру концептуальної моделі СППР. Проаналізовано можливість багаторазового використання компонентів різних типів структур. Описано принципи побудови користувацької СППР із використанням багаторазових компонент.

Ключові слова: об’єктно-орієнтований підхід, архітектура, структура, модель, прийняття рішення, система підтримки прийняття рішень.

This article describes an object-oriented approach of DSS modeling. Three-level architecture is suggested and it describes a conceptual model of DSS. The analysis of possible reusable components for various types of structures is analyzed. We describe the principles of building a custom DSS use of multiple components.

Key words: object-oriented paradigm, architecture, frameworks, model, decision making, Decision Support System.

### Вступ. Загальна постановка проблеми

Однією з постійних проблем менеджерів є розуміння можливостей та небезпек ринку і своєчасне прийняття рішень, які покращать використання корпоративних ресурсів. Системи підтримки прийняття рішень (СППР) – це комп’ютерні системи, призначені для підвищення ефективності прийняття рішень, допомоги особам, які приймають рішення щодо використання моделі і даних для вирішення частково структурованих та неструктурованих проблем прийняття рішення [1]. СППР – це сукупність інтелектуальних інформаційних застосувань та

інструментальних засобів, які використовуються для маніпулювання даними, їхнього аналізу і надання результатів такого аналізу кінцевому користувачеві. Сучасна СППР дає змогу передбачати ступінь впливу ухвалених рішень на подальший розвиток бізнесу.

Основними проблемами, які виявлено в ранніх проектах СППР, є велика споживча вартість та складність адаптації до швидких змін у вирішенні вимог підтримки організації. Швидкий прогрес у галузі інформаційних технологій сприяв дослідженню ступеня *універсальності* як основи для гнучкого та швидкого розвитку СППР.

### Аналіз останніх досліджень та публікацій

З кінця 1970-х років основними тенденціями розвитку були генератори СППР і узагальнені системи управління моделями (СУМ). В обох випадках підґрунтям універсальності для розвитку СППР є *функціонально-орієнтовані* моделі.

Легкий і швидкий розвиток комп'ютерних систем не є єдиним завданням сфери СППР. Наприклад, воно не відрізняється від основних завдань розроблення програмного забезпечення. У цій галузі повторне використання розглядається як ключ для підвищення якості та продуктивності розроблення програмного забезпечення, з основним акцентом на потенційний внесок об'єктно-орієнтованого (ОО) підходу [2].

Розглядаючи наявні концептуальні моделі СППР, можна виділити підходи, що ґрунтуються на використанні ідеології інформаційних систем, штучного інтелекту, а також інструментальний підхід [1].

Перспективним сьогодні є застосування об'єктно-орієнтованої парадигми до побудови концептуальної моделі СППР. Можливість багаторазового використання вважається ключем для досягнення продуктивності та якості програмного забезпечення. СППР – це системи, призначені для підвищення ефективності прийняття рішень, але інформаційні технології можуть істотно впливати на прийняття рішень за наявності методів, що надають можливість легко і швидко розробляти СППР. Процес розроблення СППР можна зробити ефективнішим, використовуючи предметно-орієнтовані компоненти багаторазового використання, що позитивно вплине на якість такої СППР. Для досягнення цих цілей особливим є внесок об'єктно-орієнтованої (ОО) парадигми.

Структури – це складні одиниці багаторазового використання, що складаються з класів, які спеціально призначені для удосконалення і використання як групи. Структуру [3] можна розглядати як високого рівня програму або архітектуру підсистеми, що складається з множини класів, які спеціально розроблені, щоб бути покращеними та використаними як група. Структура представляє спільні рішення для розроблення додатків у конкретній області (наприклад, графічні редактори, операційні системи), які можуть бути налаштовані відповідно до особливостей завдання. Предметно-орієнтовані компоненти, в яких структури використовуються для подання класів проблеми прийняття рішення як прототип елементів прийняття рішення, є підґрунтям вирішення проблем з розроблення та моделювання СППР [3]. Цей підхід як *інструмент моделювання* призначений для децидента (особи, що приймає рішення). Структура розглядається як загальна модель прийняття рішення для задач певного класу, яку можна налаштувати через *процес реалізації*, щоб подати особливості наявної проблеми прийняття рішення. Конкретні моделі можуть бути побудовані шляхом вибору, адаптації та об'єднання понять проблемної області.

**Не вирішені раніше частини загальної проблеми.** Модельно-орієнтовані СППР як інструмент підтримки прийняття рішень керівниками є обмеженими. Розробляють модель переважно фахівці, і менеджери часто відчують небажання використовувати модель, яку вони не зовсім розуміють і в розробленні яких вони не брали участі. Доцільно розробляти середовище моделювання для користувачів, які не є фахівцями моделювання, застосовуючи багаторазові компоненти.

### Цілі (завдання) статті

Основним завданням статті є розроблення концептуальної моделі СППР та її компонент, керуючись об'єктно-орієнтованою парадигмою. Підхід призначений для професіоналів (тобто програмістів зі знанням ОО механізму), які мають створити СППР, багаторазово використовуючи структури.

## Основний матеріал

Жодне загальне рішення не можна побудувати без розуміння характеру і характеристик конкретних дій, які воно породжує. Узгоджена функціональна архітектура СППР – це структура з трьох основних функціональних компонентів, а саме: компонента моделі (СУМ), даних (СУБД) та діалогу. СППР такої функціональної архітектури називають модельно-орієнтованими СППР.

Об'єктно-орієнтований підхід застосовують професіонали (програмісти, фахівці зі знанням ОО механізмів) для створення СППР, багаторазово використовуючи структури. Одна з основних цілей дослідження СППР – забезпечення середовища моделювання для користувача, який не є фахівцем з моделювання [4]. Основна ідея використання багаторазових компонент полягає в тому, що за допомогою додатків компонентно-орієнтованого моделювання СППР розроблення моделі для децидента стає простим процесом вибору та застосування комплексу спеціальних предметно-орієнтованих понять для опису вирішення проблемної ситуації [3]. Можна виокремити особливості таких користувацьких СППР (*DSS users*), а саме: парадигма моделювання; візуальне подання; рекомендації.

Децидент має подати свої конкретні проблеми на концептуальному рівні. Це можливо, якщо предметно-орієнтовані стандартні блоки, які використовуються для побудови СППР, пропонуються дециденту як *концепції моделювання*. Побудова моделі полягає в налаштуванні загальної моделі прийняття рішення для класів проблем прийняття рішення шляхом реалізації понять та пов'язування стандартних блоків. Наприклад, клас проблем інвестиційного бюджету ІТ-проекту. Тривалість життя проекту може продовжуватися декілька років та передбачати прогнозування всіх доходів і витрат на певний період. Рентабельність інвестицій можна оцінити за деякими критеріями, а саме: чиста поточна вартість (ЧПВ); внутрішня норма прибутку; період окупності тощо. Моделі для цього класу проблем прийняття рішень можна структурувати та сформулювати з погляду таких концепцій, як тривалість життя ІТ-проекту, графік його доходів і витрат, рух грошових коштів, критерії рентабельності, вартість капіталу тощо.

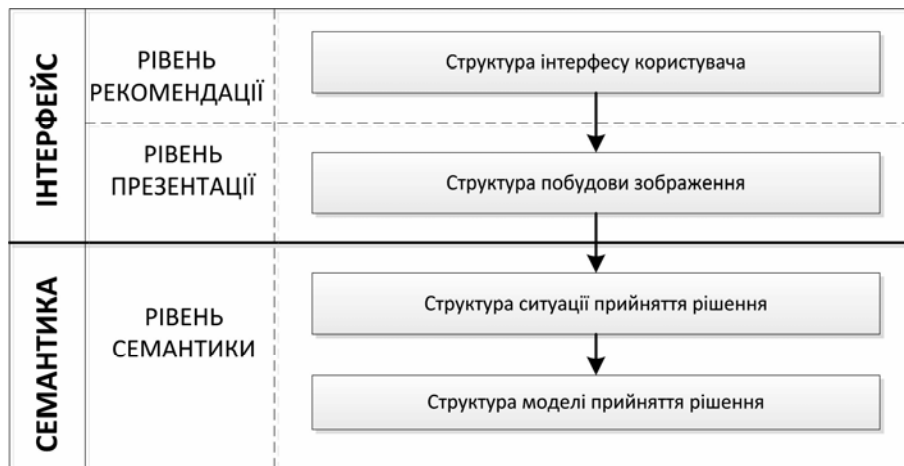
Для візуального зображення понять використовуються такі структури подання, як таблиці, графіки, звіт, діаграма тощо. Саме через це візуальне зображення користувачі можуть створювати приклади понять моделі і керувати ними.

Рекомендації для розроблення моделі даються засобами компоненту діалогу. Вони мають допомогти у виборі концепцій для розроблення моделі, а також реалізації та поєднання блоків. Рекомендації також потрібні для виконання моделі.

Враховуючи цілі СППР та особливості застосування об'єктно-орієнтованої парадигми для проектування, концептуальну модель СППР можна подати у вигляді архітектури трьох рівней, а саме: рівня семантики, презентації та рекомендації (рисунок). Ця архітектура відповідає вимогам до інтерактивних додатків, які сприяють незалежності між застосуванням семантики та інтерфейсу [5]. Різниця між рівнями в архітектурі СППР дає змогу ідентифікувати різні обов'язки, які мають набори класів, що представляють кожну частину, а саме:

- **рівень семантики:** його класи використовуються для опису спільного рішення, спрямованого на клас вирішення проблеми. Пропонуються методи моделювання проблеми прийняття рішення та її вирішення за допомогою запитань і процедур рішення;
- **рівень презентації:** його класи використовуються для опису структури загального представлення (таблиці, графіки тощо); може використовуватися для візуальною презентації концепцій, які подані на рівні семантики. Пропонуються методи для вирішення концептуального значення структури презентації, її графічного зображення;
- **рівень рекомендації:** його класи охоплюють всі аспекти розмовного інтерфейсу, що визначає взаємодію сценаріїв для моделювання та виконання заходів.

Архітектура може розглядатися як структура вищого рівня для побудови цілей СППР, які складаються з нижчого рівня типів структури і принципів їхнього поєднання. Структура діє як “спеціалізована методологія”, полегшуючи діяльність з розроблення пропозицій, детально описуючи характер і особливості різних блоків та їхній взаємозв'язок.



Концептуальна модель СППР (об'єктно-орієнтований підхід)

**Компоненти багаторазового використання рівня семантики.** Моделі рішення – це представлення елементів, які характеризують проблеми прийняття рішення, для того, щоб шукати, аналізувати та оцінювати можливі рішення. Модель рішення – це завжди неповне і наближене представлення реальності. Вона зображає *особливий підхід* до аналізу оригінальної проблеми прийняття рішення і може бути подана відповідно до послідовних *рівней абстракції*. За аналогією з галуззю розроблення програмного забезпечення моделі прийняття рішення розглядаються відповідно до трьох основних рівнів абстракції, а саме: *специфікації* (простір проблеми), *проектуювання* і *реалізації* (простір вирішення). Простір вирішення в галузі СППР стосується вибору методу чи техніки для *вирішення* проблеми. Кожна технологія вирішення передбачає *парадигму моделювання*, що складається з набору конкретних концепцій і відносин, згідно з яким має бути структуроване вирішення проблеми. Простір проблеми СППР окреслює характеристику ситуації прийняття рішення незалежно від методики її вирішення. Одним зі способів наближення до вирішення проблеми на рівні специфікації є використання наявної теорії для розв'язання цього класу задач прийняття рішення. Поняття і практики наявних теорій утворюють *домену теорії*.

Для розроблення рівня семантики СППР визначено два типи структур, а саме: структура ситуації прийняття рішення і структура моделі прийняття рішення.

*Структура ситуації прийняття рішення* забезпечує загальну парадигму розроблення моделі, яка орієнтована на певний клас вирішення проблем. Структура ситуації прийняття рішення – набір взаємопов'язаних класів, що представляють предметно-орієнтовані, цільові концепції моделювання, отримані з домену теорії. Ступінь універсальності структури ситуації прийняття рішення щодо класу вирішення проблеми залежить від *інваріантів*, які можуть бути визначені у вирішенні проблеми моделювання цього класу з концепціями і відносинами, що належать домену теорії.

У поданні цих інваріантів моделювання класи складових структур ситуації прийняття рішення є абстракцією, або:

- запозичення понять і відношень, що належать домену теорії ;
- фактор зі загальних характеристик набору концепцій для представлення вищого рівня абстракцій;
- спеціально додані для структурування цілей або для забезпечення більшої гнучкості в процесі моделювання моделі.

Основні методи дають змогу створювати, оновлювати і видаляти концепти (зразки, приклади), а також отримувати інформацію, яка пов'язана з ними. За допомогою методів визначають типові запитання для окреслення проблем цього класу. Наприклад, типовими запитаннями для вирішення проблем формування інвестиційного бюджету ІТ-проекту є “обчислення НП”, “збільшення чистої приведеної вартості ЧПВ” тощо. Більшість методів визначають запитання, які будуть абстрактними, оскільки структури ситуації прийняття рішення не залежать від структур моделі прийняття рішення.

*Структура моделі прийняття рішення* містить загальну парадигму моделювання, яку покладено в основу методу прийняття рішення, і алгоритмічні деталі цього методу. У складі парадигми моделювання набір класів, що становлять структуру моделі рішення, яка відображає основні поняття, відношення та припущення, необхідні для розроблення моделі прийняття рішення в просторі рішень. Крім того, парадигма моделювання також містить методи, які реалізують або алгоритми для виконання, аналізу чутливості та оптимізації моделей [5, 6], або інтерфейс для виконання наявних функцій СППР генераторів/додатків. Другий варіант ґрунтується на тому, що на ринку вже доступні та широкоживані інструменти методів резолюції (електронні таблиці, MS/OR, статистичні пакети тощо). Структури також можуть містити додаткові класи для представлення структур даних, необхідних для застосування алгоритмів вирішення, а також перетворення процедури [6–8]. Спрощеним прикладом структури моделі прийняття рішення є описова алгебраїчна модель (ОАМ). Вона ґрунтується на специфікації спрямованих причинно-наслідкових зв'язків серед змінних величин. ОАМ передбачає методи вирішення, де значення змінних величин або точно представлені, або виводяться зі застосуванням відносин. ОАМ можна розглядати як набір змінних величин, де кожна змінна характеризується типом, розмірністю і множиною операцій над змінною.

**Багаторазове використання компонентів рівня презентації.** Основна мета рівня *презентації* – отримати з рівня семантики поняття, які важливі для опису та вирішення проблем цього класу, надаючи користувачам СППР узгоджений і однорідний вигляд моделей, сумісних з їхнім когнітивним світом. Ця робота ґрунтується на передумові, що наявний ряд відомих базових засобів для подання структури презентації, які знайомі дециденту і можуть бути просто адаптовані та використані як візуальне представлення різних рішень проблеми. Для децидента найуживанішими структурами презентації є таблиці, форми, ієрархічні структури, графіки, (текстові) алгебраїчні формули тощо. Базові структури характеризують ряд концепцій, відносин і поведінку незалежно від їхнього конкретного використання. Вони визначають підґрунтя основних візуальних властивостей структури, а також можливості маніпулювання. Статичні та динамічні властивості класів презентації структури не впливають на семантику.

**Багаторазове використання компонентів рівня рекомендації.** Рівень *рекомендації* охоплює всі розмовні аспекти інтерфейсу “людина–машина”, що визначають взаємодію сценаріїв для розроблення моделі та виконання. Область інтерфейсу “людина–машина” є однією з найбільш плідних для розвитку компонентів багаторазового використання. Можна застосовувати як елементи низького рівня з бібліотеки класів OOPs (вікна, діалогові вікна, кнопки), так і крупніші елементи розвитку інтерфейсу користувача.

Будь-який з них підходить для розвитку рівня рекомендації, враховуючи, що він вимагає компоненти багаторазового використання для розвитку діалогу.

**Компонування структур для побудови СППР.** Досліджено процес декомпозиції, який дає змогу ізолювати структури на різних рівнях абстракції і з різними ролями на основі чіткого визначення цільової СППР.

Розроблення додатків є процесом складання, де компоненти багаторазового застосування будуть удосконалюватися і поєднуватися, щоб розробити конкретні СППР згідно з багаторівневою архітектурою, поданою вище.

Використовуємо термін складання (компонування) просто щоб підкреслити той факт, що в контексті СППР ці класи належать різним типам структур [9, 10]. На рівні семантики процес композиції може містити три аспекти: складання різноманітних структур ситуації прийняття рішення, підбір різних структур прийняття рішення, складання структур(и) ситуації прийняття рішення в межах структур(и) моделі прийняття рішення.

Метою першого виду складання є розширення меж обраної проблемної області шляхом інтеграції підзадачі. Наприклад, інвестиційний бюджет СППР можна розглядати як комплекс таких взаємопов'язаних завдань: аналіз окремих інвестиційних проектів; вибір альтернативних інвестиційних проектів за обмеженості капіталу; вартість оцінювання капіталу; вибір джерел фінансування. Кожне зі завдань представляється у вигляді структури. Можливим також є застосування компонентів багаторазового використання.

Метою складання структури моделі прийняття рішення є забезпечення якомога більшої кількості парадигм рішення, щоб впоратися з потребами структури ситуації прийняття рішення відповідно до завдань. Третій вид складання пов'язаний з тим, що структура ситуації прийняття рішення має бути завжди зібрана, принаймні структура моделі прийняття рішення, з тим щоб гарантувати, що сформульовані моделі можна проаналізувати (наприклад, виконані). Мета полягає в тому, щоб встановити відповідність між концепціями рішення та їхнім "еквівалентним" поданням відповідно до прийняття рішення, а також для автоматизації переходу від одного до іншого рішення. Основна ідея в тому, що створення екземпляра концепції рішення ініціює створення відповідного примірника на рівні розв'язку, що дає можливість встановити ланцюжок поведінки і зв'язок на кожному рівні архітектури. Аналогічно, інші операції, що характеризують моделювання поведінки об'єктів прийняття рішення, спричинять відповідне моделювання поведінки об'єктів. Крім того, запитання, які сформульовані на найвищому рівні, мають бути пов'язані з методами, які фактично викликаються виконанням процедур на рівні моделі рішення (методи рішення).

Моделі прийняття рішень мають візуальне представлення і саме через це візуальне подання користувачі можуть маніпулювати моделями (розробляти та виконувати). Відповідно до характеристик класу вирішуваних завдань і когнітивних досліджень одну або кілька структур презентації вибирають як складову її візуального подання. Структуру побудови зображення буде складено зі складових структури семантики СППР, щоб пов'язати компоненти загального призначення структури зображення. Користувачі таким чином створюють та маніпулюють примірниками концепцій моделювання за допомогою візуального подання, що їм надається. Отже, кожна операція, що застосовується для зображення об'єктів, вирішує проблему його представлення і відповідає об'єкту прийняття рішення.

Діалог для розроблення і виконання сценаріїв розробляється зі застосуванням компонентів багаторазового використання користувацького інтерфейсу. Зрозуміло, що будь-які нові класи, які потрібні для реалізації додаткових функцій (семантики або інтерфейсу СППР), що не передбачені структурами, можуть бути додані за бажанням.

### Висновки

Для досягнення поставленої мети проаналізовано застосування об'єктно-орієнтованого підходу до швидкого та легкого проектування СППР, яке ґрунтується на багаторазовому використанні структур різних типів. Запропоновано трирівневу архітектуру концептуальної моделі СППР, яка відображає розвиток компонентів багаторазового використання, що мають різний характер, а саме: структури ситуації прийняття рішень, структури моделі прийняття рішення, структури побудови зображення і структури розроблення інтерфейсу користувача. Описано типи компонентів багаторазового використання та послідовність компонування користувацької СППР. Цей підхід можна розглядати як проектування структур вищого рівня, які складаються зі структур нижчого рівня і рекомендацій з використання цих структур для розроблення конкретних СППР.

Подальші дослідження будуть присвячені дослідженню методів, моделей та інструментів для ефективнішої підтримки загальної діяльності розроблення моделі, а саме: методам видобування знань з окремих областей і сховищ для зберігання структур.

1. Верес О.М. Компоненти концептуальної моделі системи підтримки прийняття рішень / О. М. Верес // *Комп'ютерні науки та інформаційні технології: Вісник Нац. ун-ту "Львівська політехніка"*. – 2010. – № 686. – С.103–112. 2. Johnson P. *Object-oriented technology: the competitive advantage*. GEC, 9(1): 28-41, 1993. 3. Nierstrasz O. *Component-oriented software development* / O. Nierstrasz, S. Gibbs, D. Tsichritzis // *Communications of ACM*, 35(9): 160-165, Sept. 1992. 4. Becker K. *Reusable object-oriented specifications for decision support systems* / K. Becker, F. Bodart // *In: IFIP WG 8.1 Working Conference on the Object Oriented Approach in Information Systems - Quebec City, Oct. 28-31, 1991. Proceedings. North-Holland, 1991. p. 137-155*. 5. Hartson H. *Human-computer interface development : concepts and systems for its management* / H. Hartson, D. Hix // *ACM Computing Surveys*, 21(1):5-93, March 1989. 6. Dolk D.R. & Konsynski B. *Knowledge representation for model management*

systems. *IEEE TSE*, 10(6): 619-628, Nov. 1984. 7. Blanning R.W. A relational theory of model management. In: HOLSAPPLE, C.W. & WHINSTON, A.B. (eds.). *Data base management: theory and applications*. Springer-Verlag, 1987. p. 15-53. 8. MA, J. An object-oriented framework for model management. *Decision Support Systems*, 13(2): 133-149, Feb. 1995. 9. Muhana W.A. An object-oriented framework for model management and DSS development. *Decision Support Systems*, 9(2): 217-229, Feb. 1993. 10. Muhana W.A. Symms: a model management systems that supports model reuse, sharing and integration. *European Journal of Operations Research*, 72(2): 214-242, Jan. 1995.

УДК 004.932.2

М.В. Давидов

Національний університет “Львівська політехніка”,  
кафедра інформаційних систем та мереж

## ВДОСКОНАЛЕНИЙ МЕТОД СКЕЛЕТОНІЗАЦІЇ ДВОВИМІРНОЇ ОБЛАСТІ, ЩО ВРАХОВУЄ ОСОБЛИВОСТІ КОНТУРУ

© Давидов М.В., 2013

Розглянуто задачу скелетонізації двовимірної області із врахуванням особливостей контуру. Пропонується модифікація методу скелетонізації Чжана–Суна, яка використовує попередній обхід контуру для визначення його особливих пікселів. З використанням нового методу отримано скелети, стійкі до повороту області. Метод застосовано для скелетонізації зображень кисті руки в дактильній абетці.

**Ключові слова:** скелетонізація, потоншення, розпізнавання дактилю.

The problem of two-dimensional area skeletonization that takes into account boundary features is considered. A modification of Zhang-Suen parallel thinning algorithm is proposed. The modification utilizes boundary tracking for boundary features detection. Skeletons that were obtained by means of the proposed method were found to be persistent to region rotation. The method was utilized for dactyl images skeletonization and processing.

**Key words:** skeletonization, thinning, dactyl recognition.

### Вступ

Скелет двовимірної області має важливе значення для подання і розпізнавання об'єктів у різних областях комп'ютерної графіки. Зокрема, моделі на основі скелета добре зарекомендували себе під час розв'язання задач аналізу рукописного тексту [1], порівняння гнучких об'єктів [2], розпізнаванні перехожих [3]. Також моделі на основі скелета можна використовувати для розпізнавання елементів жестової мови та визначення положення тіла людини, яка жестикулює [4].

Основною властивістю скелета є збереження форми та топологічної структури об'єкта. Однак алгоритми побудови скелета чутливі до шуму та деформації межі об'єкта, що може серйозно змінити результат скелетонізації.

**Скелет** замкненої двовимірної області – це множина усіх точок області, кожна з яких має не менше ніж дві найближчі рівновіддалені точки контуру цієї області [5]. Такий скелет подають у вигляді відрізків та сегментів кривих.

Відомі методи побудови скелета поділяють на методи, які розглядають контур області як множину багатокутників, та методи, які працюють із растровим поданням області. Растрові методи називають методами *скелетонізації*, або *потоншення*, а побудований за ними скелет складається з множини пікселів та є певним наближенням скелета замкненої двовимірної області. Скелет, побудований методами скелетонізації, повинен бути завтовшки один піксель та має бути зв'язним для зв'язної двовимірної області.