

ПРО СТРУКТУРИ ДАНИХ І ЗНАНЬ ІНТЕЛЕКТУАЛЬНОГО РОЗВ'ЯЗУВАЧА ПРЯМИХ ТА ОБЕРНЕНИХ ЗАДАЧ МОДЕЛЮВАННЯ НА ОСНОВІ ІНТЕГРАЛЬНИХ РІВНЯНЬ

Розглядаються проблеми, пов'язані з побудовою інформаційного забезпечення для інтелектуалізованих систем моделювання, які дозволяють вибрати оптимальні методи розв'язання інтегральних рівнянь та оптимальні параметри цих методів.

На сучасному етапі видається актуальною проблема інтелектуалізації математичних обчислень. Йдеться, зокрема, про створення інтелектуалізованих програмних пакетів для математичного моделювання, в основі якого лежить розв'язання різних типів рівнянь: алгебраїчних, диференційних, інтегральних, інтегро-диференційних та ін.

Математичні методи розв'язання цих типів рівнянь добре відомі, і якби користувач точно знав, який саме алгоритм йому потрібен для розв'язання його задачі, особливих проблем не виникало б. Він міг би викликати потрібну процедуру, або, при достатній кваліфікації, навіть запрограмувати алгоритм, спираючись на розвинені засоби автоматизації програмування. Проблема пов'язана з тим, що користувач часто не є спеціалістом з математики і може не знати, яким методом слід розв'язувати задачу, або з якими параметрами слід використовувати той чи інший алгоритм. Вибір алгоритму може істотно залежати від характеру та числових значень вхідних даних. Нарешті, типовою є ситуація, коли не вдається формалізувати постановку задачі, і замість чіткої математичної моделі користувач має справу з розпливчастим словесним описом.

На основі аналізу сучасного стану програмних комплексів, призначених для автоматизації моделювання складних систем, можна зробити висновок, що нині більшість зазначених проблем вирішується на рівні прикладних програм. Програмісти змушені включати до прикладних моделюючих програм модулі, що розв'язують задачі підбору параметрів, оптимізації тощо. Підвищення рівня інтелектуалізованості засобів моделювання дозволить створити стандартизовані мо-

дулі, метою яких є перенесення розв'язання вказаних задач на рівень ядра моделюючого програмного комплексу.

Отже, основними цілями створення інтелектуалізованих програмних середовищ для математичного моделювання слід вважати:

- розуміння неформалізованих запитів користувача;
- підтримку бази знань про методи розв'язання певних типів задач і пошук на основі цих знань методу розв'язання конкретної задачі;
- автоматизацію вибору оптимальних параметрів методу.

Один з класичних підходів до інтелектуалізації математичних обчислень ґрунтується на функціональних моделях, що були запропоновані Е.Тиугу [1, 2]. З іншого боку, зрозуміло, що в основі інтелектуалізованого розв'язувача математичних задач можуть лежати і продукційні моделі, оскільки правило: "Для розв'язання задачі А слід застосовувати алгоритм В", - є типовою продукцією системи.

Безперечно, при інтелектуалізації програмного забезпечення для математичних обчислень у центрі уваги повинна бути не механічна реєстрація імен методів у базі знань, а пошук зручних структур даних для представлення чисельної інформації та автоматизація маніпулювання цими даними.

Раніше вже розглядалися деякі загальні підходи до автоматизації розв'язання користувачьких задач шляхом зведення їх до інтегральних рівнянь [3,4]. Зокрема, були запропоновані принципи організації функціональних мереж для розв'язання інтегральних рівнянь виду:

$$\int_a^b K(t, s) y(s) ds + y(t) = u(t). \quad (1)$$

Дана стаття присвячена конкретизації та уточненню цих принципів.

Згідно з широко поширеним підходом до структуризації знань та методів роботи з ними [5, 6], слід виділяти різні сфери знань, зокрема, предметну сферу, сферу знань про користувача, сферу знань про форми і методи ведення діалогу та ін. Необхідно підкреслити, що, коли ми визначаємо певні сфери знань, ми повинні задати і семантику зв'язків між цими сферами. Тому інтелектуалізований розв'язувач математичних задач на досить загальному рівні можна описати у вигляді набору (G_1, G_2, \dots, R) , де G_i – розв'язувачі задач з окремих сфер знань, R – формалізовані описи зв'язків між окремими сферами. Під розв'язувачем задач тут мається на увазі деяка композиція знань та металпроцедур, що дозволяють маніпулювати знаннями з метою генерації алгоритмів розв'язання користувацьких задач. Слід відмітити, що окремі сфери знань можуть перетинатися між собою.

Основним питанням, що висвітлюється у даній статті, є предметно-незалежна сфера знань про методи розв'язання інтегральних рівнянь, до яких зводяться задачі користувача. Формалізовані описи цих рівнянь можна вважати внутрішньою мовою системи. Вимога незалежності від предметної галузі впливає з того міркування, що розв'язання рівняння не повинно залежати від того, яким чином отримане це рівняння.

Інформаційну модель для опису даних зручно розглядати на двох рівнях. Як інформаційна модель верхнього рівня розглядається четвірка $\langle \Omega, \Omega^*, M, F \rangle$, де:

Ω – система породжуючих випадкових процесів; окремі функції можуть розглядатися як реалізації цих процесів;

Ω^* – визначена множина конкретних функцій;

M – сукупність модельних описів, за допомогою яких можна зображувати окремі функції; типовими є, наприклад, описи на основі ортогональних розкладів;

F – сукупність залежностей між елементами $\sigma(\Omega^* \cup M)$, $\sigma(W)$ – множина всіх підмножин множини W .

Кожний з елементів Ω, Ω^*, M, F може бути описаний у вигляді деякої фреймонічної

структури, і тоді структури даних для опису конкретних функцій, модельних просторів та самих рівнянь можна розглядати як деякі похідні фрейми, а окремі функції, модельні простори, рівняння та методи їх розв'язання – як екземпляри відповідних фреймів. Якщо деякий елемент залежить від параметрів, ці параметри природно описувати як слоти відповідних фреймів.

Функціональна мережа для розв'язання інтегральних рівнянь може розглядатися як нижній рівень дворівневої моделі, що описується. Загальні принципи організації цієї мережі наведені в [3, 4], але ці принципи вимагають подальшого розвитку.

Очевидно, для розв'язання задачі користувача в робочій пам'яті системи повинен бути сформований робочий контекст задачі, в якому мають бути визначені:

- тип задачі: пряма чи обернена;
- параметри інтегрального рівняння; основним з цих параметрів є його ядро; для ядра, в свою чергу, повинні бути задані конкретний вигляд функції $K(t, s)$ та форма її відображення; звідси випливає, що фрейм ядра має формуватися як композиція фреймів Ω^* і M ;
- початкові дані – підінтегральна функція $y(t)$ для прямої задачі або права частина $u(t)$ для оберненої задачі; легко бачити, що відповідні структури даних також повинні формуватися як композиції фреймів Ω^* і M . Більш точно, $v = L(q, m)$; де v – відображення конкретної функції, L – заданий алгоритм формування представлення, q – екземпляр фрейму Ω^* , – екземпляр фрейму M .

У [3, 4] введено два типи зв'язків між окремими компонентами функціональної мережі:

- зв'язки між описами однієї і тієї ж функції у різних модельних просторах. З наведених вище міркувань випливає, що перехід від одного опису до іншого відповідає зміні екземпляра m фреймом M , що бере участь у формуванні опису; такий перехід може бути пов'язаний з виконанням певних процедур (типовий приклад – перетворення Фур'є, яке переводить описи функцій з часової області у частотну);

- зв'язки між вхідними та вихідними функціями. Ці зв'язки реалізуються за допомогою методів розв'язання типових рівнянь. Кожний з цих методів вимагає виконання певних умов; зокрема, належність функцій до деяких класів та зображення функцій у визначеному модельному просторі. Саме остання обставина зумовлює не-

обхідність переходів від одного модельного простору до іншого.

Тоді процес розв'язання користувацької задачі на найбільш загальному рівні можна уявляти як активізацію мережних зв'язків, поки збудження не досягне деякого цільового вузла.

Типовою є ситуація, коли можна застосовувати різні методи. Крім того, необхідно здійснювати вибір параметрів методів. Тому інтелектуальний розв'язувач, крім базової функціональної мережі, повинен мати у своєму складі ще як мінімум дві компоненти:

- експертні рекомендації, що визначають, які методи краще застосовувати в даній ситуації;
- метапроцедури, що забезпечують оптимальний вибір параметрів методів.

Дуже важливим видається таке міркування. Методи реалізації переходів разом з умовами їх застосування (фактично - методи розв'язання інтегральних рівнянь) можна розглядати як продукції деякої системи, а описану функціональну мережу разом з метапроцедурами маніпулювання нею - як деяку надбудову над цією продукційною системою.

Описана ідеологія, звичайно, може бути застосована не тільки для розв'язання інтегральних рівнянь. Таке інтелектуалізоване програмне середовище може бути дуже корисним при порівняльному аналізі різних методів стиснення і обробки аудіоданих, відеозображень та іншої інформації. Крім того, воно може бути використане в навчальних цілях як ілюстрація типових методів розв'язання інтелектуальних задач.

1. *Тьугу Э.Х.* Концептуальное программирование. - М.: Наука, 1984. - 256 с.

2. *Поспелов Д.Л.* Ситуационное управление: теория и практика. - М.: Наука, 1986. - 288 с.

3. *Верлань А.Ф., Олецкий О.В.* Організація функціональної мережі для моделювання складних систем на основі інтегральних рівнянь // Збірник наукових праць Інституту проблем моделювання в енергетиці. - Львів: Світ, 1998. - Вип.3. - С 115-119.

4. *Verlan A., Oletsky A.* An object-oriented architecture of an intelligent program environment for modeling and simulation of dynamic systems // Advanced Computer Systems. Proc. of the Fifth International Conference. - Szczecin, Poland, November, 1998. - P. 244 - 248.

5. *Понов Э.В.* Общение с ЭВМ на естественном языке. - М.: Наука, 1982. - 316 с.

6. *Понов Э.В.* Экспертные системы: решение неформальных задач в диалоге с ЭВМ. - М.: Наука, 1987. - 285 с.

A.V. Oletsky

THE STRUCTURES OF DATA AND KNOWLEDGE FOR INTELLECTUAL SOLVERS OF DIRECT AND INVERSE SIMULATION TASKS BASED ON INTEGRAL EQUATIONS

Problems concerning intellectual simulation packages are regarded. These packages should be able to choose proper methods for solving integral equations and optimal parameters of these methods.