

бути досить перспективним. До таких задач можна віднести задачу комп'ютерної реалізації інтегрального перетворення Карунена-Лоева. Це перетворення є оптимальним для стиску інформації за критерієм середньоквадратичної похибки. На практиці в основному використовується дискретний варіант цього перетворення, що не є адекватною апроксимацією інтегрального, і тому його використання в задачах, що вимагають точних моделей сигналів, не завжди є ефективним.

Практичне застосування перетворення Карунена-Лоева стримується відсутністю для нього швидких перетворень. Проте це перетворення може використовуватися для задач, для яких швидкість не є критичною, але вирішального значення набуває саме стиску інформації. До таких задач можна віднести задачу ефективного зберігання даних в системах обробки інформації та в мультимедіа-системах. Зокрема, перспективною видається розробка спецпроцесорів та програмного забезпечення для інформаційних перетворень, що покращили б нинішні стандарти - JPEG та MPEG.

Інша проблема — розв'язання задачі відновлення сигналів, спотворених зовнішніми шумами або низькоякісною апаратурою. Можна довести, що існують лінійні перетворення, що забезпечують абсолютно точне відтворення потрібного сигналу. Але, на жаль, цей факт не може бути безпосередньо використаний для вирішення проблеми, оскільки вигляд цього перетворення суттєво залежить від невідомих величин. Тут можуть бути використані подібні адаптивні самонавчальні процесорні системи.

ЗАСТОСУВАННЯ КВАЛІФІКАЦІЙНИХ ФУНКЦІЙ ДЛЯ ОЦІНКИ ІНФОРМАЦІЙНИХ РЕСУРСІВ ПРОГРАМНИХ СИСТЕМ

С. Спасітєлева (кафедра інформатики НаУКМА)

Зручним засобом формального аналізу поведінки програм є кваліфікаційна функція [1]. Поведінку програмної системи можна описати з використанням двох часткових видів кваліфікаційних функцій - трансформаційних і часових. При цьому можна описати і отримати оцінку інформаційних ресурсів, зокрема, процесорного часу, якого потребує кожний програмний модуль системи. Трансформаційні функції віддзеркалюють значення змінних під час виконання програм, а часові функції символічно виражають час виконання програм.

Пропонується формальний механізм, який базується на кваліфікаційних функціях [1] і є основою для опису поведінки систем реального часу та алгоритмів побудови часових формул. Виходячи із [1], введемо означення трансформаційних та часових функцій.

Означення 1. Нехай \vec{X} - вектор змінних програми, S - оператор у програмі, I - скінченна множина, $\{B_j\}$ - множина логічних умов, які визначають наступні гілки у графі виконання S (мал.1). Тоді трансформаційна функція $F_S(\vec{X})$ визначає новий вектор \vec{X} після виконання оператора S

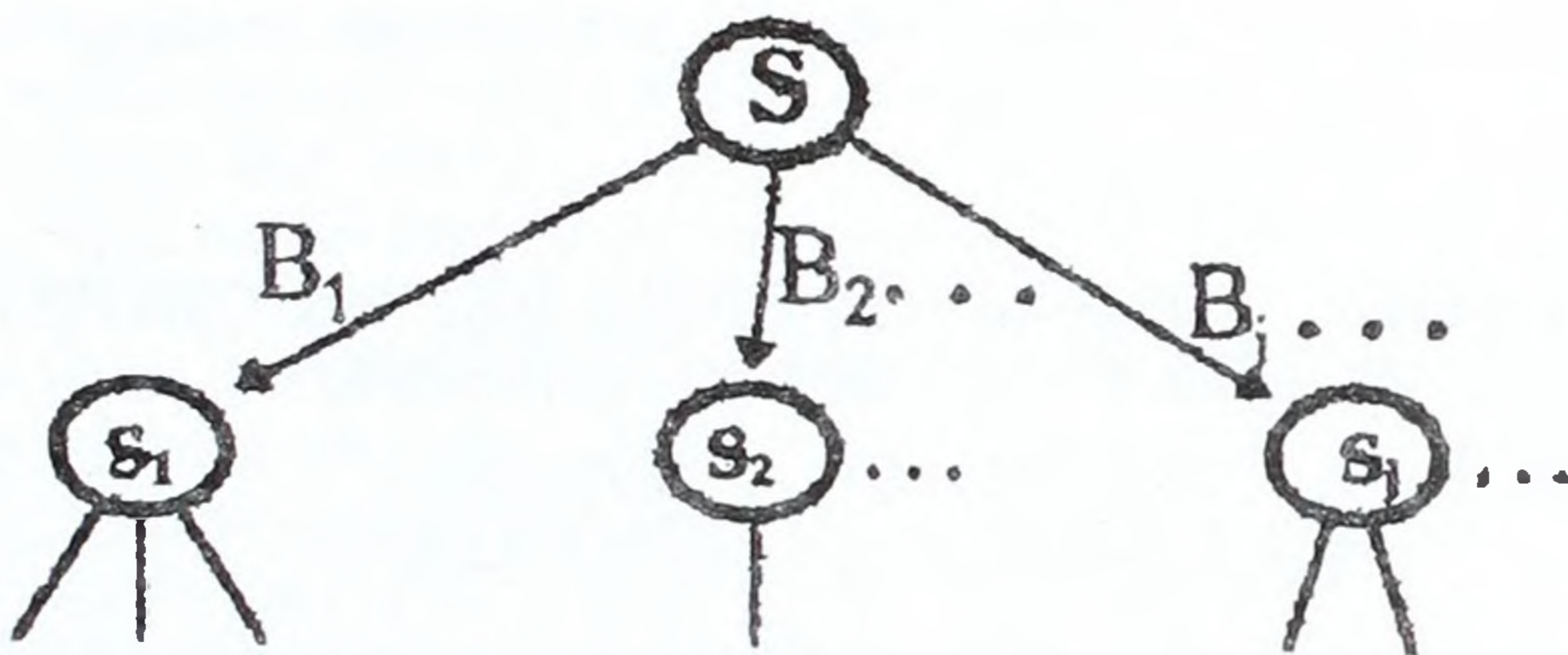
$$F_S(\vec{X}) = \left\{ (B_j, F_j) \right\}_{j \in J},$$

де F_j - трансформаційна функція, пов'язана з оператором S_j .

Тип (R_1, E_1) кваліфікаційної функції F_S для розглядуваних детермінованих програм досить простий і визначається таким чином:

R_1 (обмеження): існує лише одне $j \in J$, при якому B_j набирає значення "Істина";

E_1 (спосіб обчислення): нехай $j \in J$ - єдиний елемент, при якому $B_j(\vec{X})$ набирає значення "Істина", тоді $F_S(\vec{X}) = F_j(\vec{X})$. Якщо $F_j(\vec{X})$ - простий вектор, то $\tilde{F}_j(\vec{X}) = F_j(\vec{X})$.



Мал. 1.

Означення 2. Нехай S - оператор програми, I - скінченна множина, \vec{X} - вектор змінних до виконання оператора S . Тоді $T(S:\vec{X})$ - часова функція виконання оператора S з

вхідними змінними $\vec{X} : T(S:\vec{X}) = \left\{ (C_i, T(S_i:\vec{X})) \right\}_{j \in J},$

де $\{ C_i \}$ - множина логічних умов, які визначають наступні гілки у графі виконання складного оператора S (мал.1), $T(S_i: \bar{X})$ — часова функція, пов'язана з виконанням оператора $S_i S$.

Тип (R_2, E_2) кваліфікаційних функцій, які віддзеркалюють часові функції, визначається таким чином:

обмеження R_2 відсутнє, оскільки приймається довільна множина $\{ C_i \}$;

спосіб обчислення E_2 задається формулою:

$$\tilde{T}(S: \bar{X}) = \sum_{C_i = \text{істина}} \tilde{T}(S_i: \bar{X}),$$

де $\tilde{T}(S_i: \bar{X}) = T(S_i: \bar{X})$, якщо $T(S_i: \bar{X})$ -

арифметичний вираз.

Також пропонуються правила для побудови часових функцій та правила виведення часових функцій для довільних програмних конструкцій. На цій основі будуються алгоритми часового аналізу програмних систем.

Література:

1. *Gabrielian A., McNamee L., Trawick D.* The qualified function approach to analysis of program behavior and performance. //IEEE Transactions on software Engineering. - 1985. - SE-11, №8. -P.758-773.

ТРЬОХРІВНЕВА МОДЕЛЬ ІНФОРМАЦІЙНИХ СИСТЕМ В АРХІТЕКТУРІ КЛІЄНТ-СЕРВЕР

С. Спасітелева, І. Чичкань
(кафедра інформатики НаУКМА)

В основу трьохрівневої моделі покладено принцип, згідно з яким декомпозиція великих складних задач не тільки спрощує, а і прискорює їх виконання. В однорівневій моделі все, що відноситься до додатків, за винятком використовуваних ними зовнішніх даних, розташовано на локальній робочій станції.

У найбільш популярній двохрівневій моделі логіка обробки даних залишається на комп'ютері користувача, а операції з базою даних (БД) здійснюються на тому ж сервері, на якому ця БД зберігається.