

На правах рукописи

Пасечников Константин Алексеевич

**СИНТЕЗ ОПТИМАЛЬНЫХ
СТРУКТУР ДАННЫХ ДЛЯ АЛГОРИТМОВ
РЕШЕНИЯ КОМБИНАТОРНЫХ ЗАДАЧ НА ГРАФАХ**

**Специальность 05.13.11 – Математическое и программное обеспечение
вычислительных машин, комплексов и компьютерных сетей**

**Автореферат
диссертации на соискание ученой степени
кандидата технических наук**



Москва - 2009

Работа выполнена в Московском государственном техническом университете им. Н.Э. Баумана

Научный руководитель: доктор технических наук, доцент
Иванова Галина Сергеевна

Официальные оппоненты: доктор технических наук, профессор
Жданов Владимир Сергеевич

кандидат технических наук
Башмаков Александр Игоревич

Ведущая организация **Научно-технический инновационный центр «ТЕХКОМ»**

Защита диссертации состоится «04» июня 2009 г. в 16:00 часов на заседании диссертационного совета Д 212.141.10 в Московском государственном техническом университете им. Н.Э. Баумана по адресу:
105005, 2-я Бауманская ул., д. 5.

Ваши отзывы в двух экземплярах, заверенные печатью, просьба высылать по указанному адресу.

С диссертацией можно ознакомиться в библиотеке Московского государственного технического университета им. Н.Э. Баумана.

Автореферат разослан « » 2009 г.

Ученый секретарь
диссертационного совета,
к.т.н., доцент



Иванов С.Р.

ОБЩАЯ ХАРАКТЕРИСТИКА РАБОТЫ

Актуальность темы.

Большинство алгоритмов решения задач структурного синтеза на графах являются NP -сложными, а сами объекты особенно применительно к вычислительной технике имеют большую размерность. В настоящее время речь идет о сотнях тысяч компонентов и более – при исследовании и разработке структур аппаратных средств и о тысячах компонентов – для задач, связанных с созданием программного обеспечения. Соответственно время решения многих задач рассматриваемого класса даже на современной технике недопустимо велико.

Исследованиями в области решения задач структурного синтеза, а также проблемами уменьшения вычислительной и емкостной сложности применяемых алгоритмов решения этих задач занимались А.В. Ахо, Т. Кормен, Ч. Лейзерсон, Р. Риверст, Р. Седжвик, Д. Хопкрофт, В.А. Овчинников, А.И. Петренко и многие другие. В работах перечисленных ученых показано, что вычислительная сложность вариантов реализации алгоритма при использовании различных структур данных для представления графовых моделей может отличаться на несколько порядков.

В системах автоматизации программирования при решении задач рассматриваемого класса выбор структур данных для представления графовых моделей осуществлялся по результатам анализа ограниченного набора имеющихся. Задача формализованного синтеза комбинированных структур данных до настоящего времени не была решена.

Создание средств автоматического синтеза и оценки характеристик комбинированных структур данных позволит получать оптимальные с точки зрения вычислительной сложности структуры и, следовательно, снизить вычислительную сложность алгоритмов и время их разработки.

В связи с этим синтез оптимальных структур данных для представления графовых моделей является актуальной задачей.

Цель диссертационной работы. Создание комплекса математических, лингвистических и программных средств для автоматической генерации структур данных, оптимальных с точки зрения вычислительной сложности операций над данными и используемых для представления объектов задач структурного синтеза.

Задачи исследований. Для достижения указанной цели потребовалось:
Для достижения указанной цели в работе потребовалось:

1. Выполнить анализ существующих структур данных с целью выявления их свойств и классификации основных операций над ними.
2. На основании проведенного анализа разработать модели структур данных, позволяющие оценивать влияние организации данных на емкостную и вычислительную сложность алгоритма.

3. Выполнить формальную постановку задачи синтеза оптимальной структуры данных и разработать метод её решения.
4. Разработать способ генерации синтаксически корректных конструкций описаний оптимальных структур данных.
5. Разработать программную систему синтеза оптимальных структур данных для представления взвешенных множеств и графовых моделей.
6. Выполнить экспериментальную проверку полученных теоретических результатов.

Методы исследований. Результаты диссертационной работы получены на базе использования методов анализа, синтеза и оптимизации. Математическую основу составляет аппарат теории графов, теории множеств и математической логики, а также теория формальных языков и методы грамматического разбора.

Научная новизна работы. В процессе исследования получены следующие новые научные результаты, представляемые на защиту:

1. Выполнена систематизация существующих и предложены новые способы оптимизации структур данных, представляющих графовые модели, по критерию минимальной вычислительной сложности операций над ними.
2. Разработаны структурные модели базовых и комбинированных способов организации данных, применимые для автоматической оценки вычислительной и ёмкостной сложности алгоритмов решения задач на графах, а также определены формальные правила перехода от объектов к моделям.
3. Формально определена операция объединения моделей структур данных, позволяющая получать модели комбинированных структур данных и их характеристики из базовых.
4. Получена формальная постановка задач синтеза оптимальных с точки зрения минимума вычислительной сложности алгоритма одноуровневых и двухуровневых структур данных для представления графовых моделей, разработаны и реализованы алгоритмы их решения.
5. Разработан алгоритм генерации синтаксически корректных конструкций описаний структур данных для синтезированных моделей.

Практическая ценность работы. По результатам проведенных исследований создана программная система синтеза оптимальных структур данных, которая включает модуль генерации оптимальных структур данных, анализатор вычислительной и ёмкостной сложности структур данных, библиотеки шаблонов описаний классов абстракций, а также макрогенератор описаний структур данных. Система позволяет автоматизировать процесс

синтеза структур данных, а также позволяет получать оценки вычислительной и ёмкостной сложности для сгенерированных структур.

Апробация работы. Основные положения работы обсуждались на научно-технической конференции, посвященной 175-летию МГТУ им. Н.Э. Баумана (Москва, 2005), на научной конференции «Информатика и системы управления», проходившей в МГТУ им. Н.Э. Баумана (Москва, 2007).

Реализация и внедрения. Все основные теоретические и практические результаты работы в виде методики и программных средств внедрены в промышленность. Они использованы при разработке очередной версии системы защиты Windows-приложений StarForce Protection Builder 5.70 в ООО «Протекшен Технолоджи» (StarForce) и при создании системы анализа и классификации текстов БКФ 2.00 в ЗАО «ИнфоВотч» (дочерняя компания ЗАО «Лаборатория Касперского»).

Публикации. По результатам диссертационной работы автором опубликовано 7 статей.

Объем и структура диссертации. Диссертационная работа включает введение, четыре главы, заключение и список литературы, занимающих 159 страниц текста, в том числе 32 рисунка и 23 таблицы, список использованной литературы из 58 наименований на 6 страницах.

СОДЕРЖАНИЕ РАБОТЫ

Во введении обоснована актуальность темы диссертации, рассмотрено современное состояние проблемы. Сформулированы цель и основные задачи исследования, приведен перечень основных научных результатов, выносимых на защиту, и изложено краткое содержание глав диссертации.

В первой главе «Анализ существующих методов синтеза структур данных и постановка задачи» выполнен анализ существующих способов оптимизации структур данных и показана возможность их применения в случае автоматического синтеза структур данных.

Предложены три способа специализации структур данных: статическая, динамическая и адаптивная специализация. Данные способы позволяют адаптировать программу, использующую абстрактные типы данных, к отдельным вариантам использования посредством генерации структур данных, оптимальных для данного варианта использования.

Проведён анализ существующего оптимизатора языка SETL – единственной на данный момент программной системы, решающей задачу выбора оптимальных структур данных. Анализ оптимизатора выявил 6 существенных недостатков, которые накладывают значительные ограничения на его применение.

Выполнен анализ формального описания структур данных, применяемого в языке АЛГОРИТМ. Показано, что использование предлагаемых мо-

делей структур данных для решения задачи синтеза оптимальной структуры данных неэффективно вследствие высокой сложности последних.

Содержательно определены задачи, которые необходимо решить при создании системы генерации оптимальных структур данных. К ним относятся задачи разработки моделей комбинированных и многоуровневых структур данных, формальной постановки задачи синтеза оптимальных структур данных, разработки синтезатора оптимальных структур данных для представления графовых моделей и макрогенератора описаний структур данных.

Во второй главе «Разработка моделей структур данных» выполнен анализ наиболее часто используемых операций над структурами данными и предложена классификация, позволяющая определить набор операций, который можно использовать при сравнении двух структур данных по временным параметрам. Этот набор состоит из операций, вычислительная сложность которых не зависит от используемых инструментальных средств (компиляторов и операционных систем) и включает в себя элементарные операции анализа и преобразования.

Разработаны модели структур данных, отражающие структурные особенности различных способов организации данных, существенные с точки зрения вычислительной сложности выполняемых над структурой данных операций, а также ее ёмкостной сложности.

Моделью **векторного способа организации данных** является смешанный граф (рис. 1). Переход от объекта к модели для вектора выполняется по следующим правилам:

- 1) Любая структура данных представляет собой описание одного из возможных способов размещения конечного множества данных D в памяти. При этом каждому элементу множества данных ставится во взаимнооднозначное соответствие P адрес из множества доступных адресов памяти $A \leftrightarrow P \subset \{ \dots \}$. При переходе от объекта к модели поставим во взаимнооднозначное соответствие множеству D множество вершин графа X_D , а множеству адресов A – множество вершин графа X_A . Поскольку множество P описывает взаимнооднозначное соответствие между множествами A и D , каждой паре элементов множества P сопоставляем ребро $u_{AD} \in E_{AD}$.
- 2) Структура данных кроме полезной информации в виде элементов данных обычно хранит служебную информацию о связях элементов между собой. Элемент данных вместе со служебной информацией назван *компонентом* структуры данных. Таким образом, можно говорить о наличии множества C компонентов структуры данных и множества Z , определяющего взаимнооднозначное соответствие размещенных элементов данных компонентам структуры $P \leftrightarrow Z \subset \{ \dots \}$. Компонентом вектора будет являться непосред-

венно элемент данных, а множество связей компонентов F будет представлять собой декартово произведение множества C самого на себя, т. е. $F = C \times C$. Таким образом, каждому компоненту $c \in C$ вектора будет взаимнооднозначно соответствовать двухвершинный подграф с одним ребром $g(\langle X_A, X_D \rangle, U_{AD}) \in G$, где G – граф, описывающий организацию элементов структуры данных.

- 3) Связи F между компонентами структуры данных отобразим, определив однозначное соответствие $C \rightarrow F \subset C \times C$. Каждой связи компонентов из множества F сопоставим дугу из множества U_F .

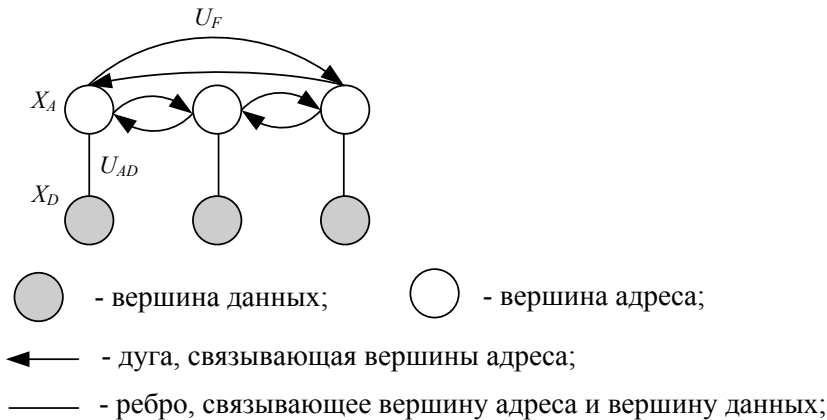


Рис. 1. Модель вектора

В результате модель вектора:

$$M_V = \langle G, Q, L(n) \rangle$$

$$Q = q_i(n) : s_i \rightarrow \overline{S}, i = \overline{|S|},$$

где $G_V(\{X_D, \langle X_A, l^V \rangle, \{U_{AD}, \overline{U_F}\})$ – смешанный граф, описывающий организацию элементов,

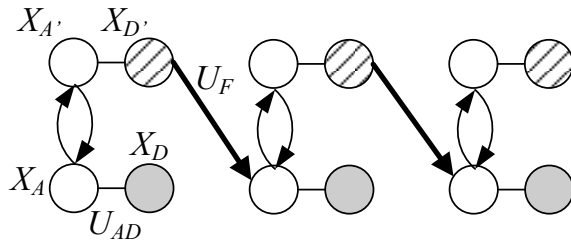
Q – множество функций, определяющих вычислительную сложность операций,

$S = \{ s_i \}$ – множество допустимых операций над структурой данных,

$q_i(n)$ – функция оценки вычислительной сложности i -ой операции,

$L(n) = \dots$ – дополнительная ёмкостная сложность структуры данных (функция, определяющая объём памяти, занимаемый вспомогательными элементами структуры данных в зависимости от количества хранимых элементов данных).

Построена модель **списковой организации данных** подразумевающей размещение элементов списка в произвольных свободных участках памяти (см. рис. 2). Сопоставление модели списка с объектом выполняется по таким же правилам, как и в случае вектора, за исключением того, что в случае списка необходимо учитывать явную связь компонентов между собой, для чего добавляются дополнительные конструкции $(X_A, X_D, U_{AD}, \overline{U_A})$



Условные обозначения:

- - вершина данных; ● (with diagonal lines) - вспомогательная вершина;
- - вершина адреса;
- ← (curved) - дуга, связывающая вершины адреса;
- (straight) - ребро, связывающее вершину адреса и вершину данных;
- ← (thick) - дуга, связывающая компоненты списка между собой

Рис. 2. Модель односвязного списка

По аналогии с вектором, модель списка будет выглядеть следующим образом:

$$M_S = \langle G_S, Q, L(n) \rangle$$

где $G_S(\{X_D, X_{D'}, \langle X_A, l^V \rangle \langle X_A, l^A \rangle, \{U_{AD}, U_{AD'}, \overline{U_A}, \overline{U_F}\})$ – смешанный граф, описывающий организацию элементов в структуре данных,

Q – множество функций, определяющих вычислительную сложность операций,

$$L(n) = nl^A - \text{дополнительная ёмкостная сложность списка.}$$

Полученные модели базовых структур данных позволяют не только описывать организацию элементов данных, но и задавать вычислительные и ёмкостные параметры структур данных, также предложенные модели являются более простыми, чем модели, используемые в языке АЛГОРИТМ, так как они не содержат избыточной, с точки зрения поставленной задачи, информации.

Разработаны модели комбинированных структур данных и определены формальные правила, позволяющие получать комбинированные структуры данных из базовых. Для иллюстрации формальных правил объединения рассмотрим объединение древовидного списка M_1 и вектора M_2 , которые описываются следующим образом:

$$M_1 = \langle \overline{f}_1, Q_1, L_1(n) \rangle$$

$$G_1 = \overline{f}(X_1, U_1) = \overline{f}(\{X_D, X_{D1}, \langle X_A, l^V \rangle \langle X_{A1}, l^A \rangle, \{U_{AD}, U_{AD1}, \overline{U_{F1}}, U_1'\}\}),$$

$$M_2 = \langle \overline{f}_2, Q_2, L_2(n) \rangle$$

$$G_2 = \overline{f}(X_2, U_2) = \overline{f}(\{X_D, \langle X_A, l^V \rangle, \{U_{AD}, \overline{U_{F2}}\}\}),$$

где G_1, G_2 – графы организации элементов древовидного списка и вектора соответственно,

Q_1, Q_2 – множества функций, определяющих вычислительную сложность операций над исходными структурами данных,

$L_1(n), L_2(n)$ – дополнительные объёмы памяти для организации древовидного списка и вектора соответственно,

X_D, X_A – множество вершин, соответствующих элементам данных и их адресам,

U_{AD} – множество дуг, отражающих взаимнооднозначное соответствие между элементами данных и их адресами,

X_{D1}, X_{A1}, U_{AD1} – множества вершин и дуг, описывающие дополнительные поля в компонентах древовидного списка,

$\overrightarrow{U_{F1}}, \overrightarrow{U_{F2}}$ – множества дуг, отражающих связи компонентов структуры данных между собой,

U_1' – множество дуг, отражающие связи между полями внутри компонента.

Графы G_1 и G_2 имеют общие вершины (X_A, X_D) и ребра (U_{AD}), что позволяет выполнить операцию объединения этих графов. Результат объединения представлен на рис. 3.

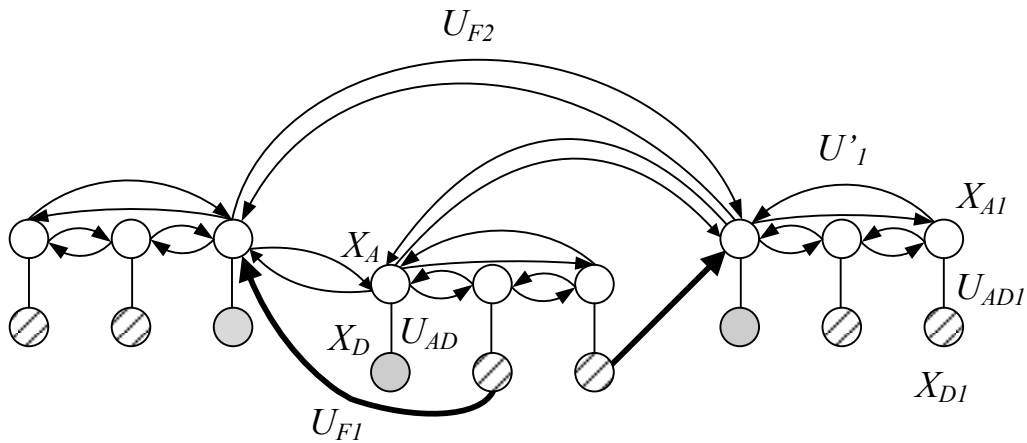


Рис. 3. Результат объединения моделей древовидного списка и вектора

Модель результирующей структуры данных:

$$M = \langle G^*, Q^*, L^*(n) \rangle$$

где $G^* = \mathcal{F}_1 \cup \mathcal{F}_2 = \mathcal{F}(\{X_D, \langle X_A, I^V \rangle, X_{D1}, \langle X_{A1}, I^A \rangle, \{U_{AD}, U_{AD1}, \overrightarrow{U_{F1}}, U_1', \overrightarrow{U_{F2}}\}\})$,

$$L(n) = l_1(n) + l_2(n),$$

$$Q^* = \mathcal{Q}_{S1} \cup \mathcal{Q}_{S2},$$

$$\mathcal{Q}_{S1} = q_i(n) = \min[q_{1i}(n), q_{2i}(n)],$$

$$\mathcal{Q}_{S2} = q_i(n) = l_{1i}(n) + l_{2i}(n),$$

$\mathcal{Q}_{S1}, \mathcal{Q}_{S2}$ – множества функций, определяющих вычислительную сложность операций анализа и преобразования соответственно.

Разработаны различные варианты объединения структур данных, содержащих указатели, а также предложен способ снижения вычислительной сложности операций преобразования над результирующей структурой, за счёт введения дополнительной связи между компонентами.

Полученные формальные правила объединения моделей структур данных позволяет синтезировать модели сложных комбинированных структур данных, а также дают возможность автоматически оценивать вычислительную сложность выполнения операций над ними и их емкостную сложность.

Выполнена формальная постановка задачи синтеза оптимальных одноуровневых структур данных. Исходными данными для синтеза оптимальной, с точки зрения вычислительной сложности заданного алгоритма, структуры данных являются:

- 1) описание множества и размер одного элемента этого множества;
- 2) совокупность базовых структур данных $B = \langle G_i, Q_i, L_i(n) \rangle$;
- 3) совокупность операций, выполняемых над множеством, что позволит определить набор операций над данными структуры;
- 4) количество повторений каждой операции над множеством в алгоритме (множество функций $R = \{r(n): S \rightarrow \dots\}$);
- 5) допустимый объем памяти $E_{\text{доп}}(n)$, который может занимать синтезируемая структура.

Отсюда получена формальная постановка задачи в следующем виде:

Найти

$$G_{\text{opt}} \in \mathcal{G} : Q_{\Sigma}(G_{\text{opt}}) = \min\{Q_{\Sigma}(G_i) / G_i \in \mathcal{G}\},$$

при

$$L(G_{\text{opt}}) \leq E_{\text{доп}},$$

$$L_{\text{доп}} = E_{\text{доп}}(n) - \nu l^V,$$

где

$$G = \{B_1, B_2, B_K\},$$

$$B_K = \{b_j \cup b_k / b_j, b_k \in \mathcal{B}, j \neq k\}, B = \{B_1, B_2, B_3\},$$

$$Q_{\Sigma}(G_i) = \sum_{s \in \mathcal{S}} q_s,$$

$$q_s = \begin{cases} q_{sj}, b_j \in \{B_1, B_2\} \\ f_s(q_1, q_2), b_j \in B_K \end{cases}$$

здесь G - полное множество возможных вариантов одноуровневых структур данных;

B - полное множество базовых структур данных (см. пункт 2 списка исходных данных);

B_K - множество всех возможных комбинированных структур данных;

$p_s \in \mathbb{R}^+$ - функция, определяющая количество повторений операции типа $s \in S$ в зависимости от размера входа задачи;

$q_s \in \mathbb{R}^+$ - оценка вычислительной сложности операции типа s над результирующей структурой данных;

$f_s(q_1, q_2)$ - функция оценки вычислительной сложности операции типа s над комбинированной структурой данных, зависящая от типов объединяемых структур и типа операции.

В этом случае задача определения оптимальной структуры данных заключается в нахождении такого способа организации элементов данных G_{opt} , при котором $Q_\Sigma \rightarrow \min$ при выполнении условия $L^*(G_{opt}) \leq L_{don}$.

Получены модели двухуровневых структур данных, используемых для хранения исходных данных задач структурного анализа и синтеза на графах. К таким структурам данных относятся матрицы, массивы массивов, массивы списков, списки массивов и т.п.

Выполнена формальная постановка задачи синтеза оптимальных двухуровневых структур данных. Исходными данными для синтеза оптимальной двухуровневой структуры данных являются все исходные данные, рассматриваемые при решении задачи синтеза оптимальной одноуровневой структуры данных, а также множество базовых двухуровневых структур данных $B^2 = \{b_{2_i} = \langle G_i, Q_i, L_i(n) \rangle\}$.

Отсюда получена формальная постановка задачи синтеза оптимальной двухуровневой структуры данных в следующем виде:

$$\text{Найти } G_{opt}^2 \in B^2 : Q_\Sigma(G_{opt}^2) = \min\{Q_\Sigma(G_i^2) / G_i^2 \in B^2\},$$

при

$$L(G_{opt}^2) \leq L_{don},$$

$$L_{don} = E_{don}(n) - \alpha^V - \beta^k,$$

где

$$B^1 = \{B_1, B_2, B_K\},$$

$$G^2 = \{g_j^2 = \langle \gamma_k^2, G^1 \rangle, b_k^2 \in B^2\},$$

$$G^1 = \{b_j^1 / b_j^1 \in B^1\},$$

$$B_K = \{b_j \cup \dots / b_j, b_k \in B, j \neq k\}, B = \{B_1, B_2, B_3\},$$

$$Q_\Sigma(G_i^2) = \sum_{s \in S} q_s * p_s,$$

$$q_s = f_s(b_k^2, G^1),$$

здесь G^2 - множество всех возможных двухуровневых структур данных;

B^2 - полное множество базовых двухуровневых структур данных;

G^1 - множество одноуровневых структур данных, включённых в двухуровневую;

B^l - множество всех возможных одноуровневых структур данных;
 $p_s \in \mathbb{N}$ - функция, определяющая количество повторений операции типа $s \in S$ в зависимости от размера входа задачи;
 q_s - оценка вычислительной сложности операции типа s над результирующей структурой данных.

В этом случае задача определения оптимальной структуры данных заключается в нахождении такого способа организации элементов G_{opt}^2 , при котором $Q_{\Sigma} \rightarrow \min$ при выполнении условия $L(G_{opt}^2) \leq L_{don}$.

Таким образом, в данной главе сформулированы основные положения, положенные в основу автоматизированной системы синтеза оптимальных структур данных.

В третьей главе «Разработка методики синтеза оптимальных структур данных и генерации их описаний» разработан алгоритм решения задачи синтеза оптимальной одноуровневой структуры данных. Результатом данного алгоритма является структура данных с минимальной суммарной вычислительной сложностью для определённого множества операций. Эта структура может быть базовой либо комбинированной, то есть полученной в результате объединения нескольких базовых структур данных.

Показано, что применение комбинированных структур данных, полученных в результате объединения более чем двух базовых структур данных, является нецелесообразным. Это обусловлено тем, что при объединении трёх и более структур наблюдается существенное увеличение вычислительной сложности операций преобразования без какого-либо уменьшения вычислительной сложности операций доступа.

Обоснована возможность использования метода полного перебора для решения задачи синтеза оптимальных одноуровневых структур данных.

Содержательное описание полученного алгоритма синтеза оптимальной одноуровневой структуры данных выглядит следующим образом:

1. Посредством применения операций преобразования к базовым структурам данных из множества B_1 (множество базовых структур данных) формируем множества B_2 (множество базовых структур данных, содержащих указатели на элементы данных) и B_3 (множество базовых структур данных, содержащих указатели на элементы данных и дополнительные указатели для связи компонентов в результирующей структуре).
2. Формируем полное множество базовых структур данных, т.е. $B = B_1 \cup B_2 \cup B_3$.
3. Определяем множество решений как совокупность всех возможных объединений всех структур данных из полного множества базовых структур и множества простых базовых структур данных, т.е.

$$R = R_1 \cup \{r_1\} \cup \{r_2\},$$

$$R_1 = \{r_{ij}, r_{ij} = \{j\} \cup \{k\}, b_j, b_k \in \{ \} \}.$$

4. Задаём начальные значения для минимальной суммарной вычислительной сложности, текущего множества решений с минимальной суммарной вычислительной сложностью: $Q_{\min} = \emptyset$, $R_{\min} = \emptyset$

5. Для $\forall r_i \in R, r_i = \langle G, Q, L(n) \rangle$ выполняем:

5.1. Определяем суммарную вычислительную сложность:

$$Q_{\Sigma} = \sum_{s \in S} (n) * \tau_s(n).$$

5.2. Проверяем условие:

$$Q_{\Sigma} \leq Q_{\min}.$$

Если условие выполняется, то переходим к п. 5.3, иначе – к п. 5.

5.3. Если установлено ограничение на максимальный размер непрерывного блока, т.е. $m \neq \infty$, то проверяем условие:

$$m \leq f_{block}(r_i, n_{\max}, l^V),$$

где f_{block} - функция, возвращающая максимальный размер непрерывного блока для заданной структуры данных при определённом количестве элементов и размере одного элемента.

Если условие не выполняется, то переходим к п. 5, иначе – к п. 5.4.

5.4. Проверяем условие:

$$E_{oon}(n) \geq \lfloor \frac{m}{l^V} \rfloor + l^V$$

Если условие не выполняется, значит, данная структура данных не удовлетворяет ограничению по ёмкостной сложности, переходим на п. 5, иначе – к п. 5.5.

5.5. Проверяем условие:

$$Q_{\Sigma} = Q_{\min}.$$

Если условие выполняется, то переходим к п. 5.7, иначе – к п. 5.6.

5.6. Очищаем множество решений с минимальной суммарной вычислительной сложностью:

$$R_{\min} = \emptyset.$$

5.7. Включаем r_i в R_{\min} :

$$R_{\min} = R_{\min} \cup \{r_i\}$$

и возвращаемся к пункту 5.

6. Проверяем условие:

$$R_{\min} = \emptyset$$

Если условие выполняется, значит, подобрать структуру данных, соответствующую заданным ограничениям, не удалось. Переходим на п. 9.

Если условие не выполняется, переходим на п. 7.

7. Ищем структуру данных с минимальной дополнительной ёмкостной сложностью, для которой одновременно выполняется условие:

$$M_{\min} = r_i \in R_{\min} : L_i(n) \rightarrow \text{in}.$$

8. Запускаем процедуру генерации описания структуры данных.

9. Конец работы алгоритма.

Некоторые входные данные, метаданные, а также результаты работы алгоритма синтеза представляют собой функции одного аргумента которые, как правило, описывают зависимость какого-либо параметра от размера входа задачи. В работе предложен способ задания функций одного переменного, а также разработаны соответствующий язык и внутреннее представление, обеспечивающие простоту распознавания и высокую скорость выполнения основных операций.

Предложена реализация операции объединения одноуровневых структур данных, позволяющая минимизировать количество разрабатываемых шаблонов классов, а также исключить необходимость объединения структур данных на уровне отдельных операций (что привело бы к необходимости учитывать возможность объединения каждой базовой структуры с любыми другими базовыми структурами).

Предложенные процедуры макрогенерации описаний структур данных на основании шаблонов обеспечивают простоту внесения изменений в описания базовых структур данных (без необходимости перекомпиляции), а также высокую скорость работы макрогенератора (см. рис. 4).

Разработан алгоритм генерации описания комбинированных структур данных.

Для задачи синтеза оптимальной двухуровневой структуры данных предложен способ сокращения размера множества решений с $6 \cdot 10^{10}$ до $2 \cdot 10^3$ вариантов, что позволило использовать метод полного перебора. Сокращение стало возможным за счёт использования свойства двухуровневых структур данных, которое заключается в том, что любую операцию над двухуровневой структурой данных можно представить в виде совокупности операций над одноуровневыми структурами, входящими в состав двухуровневой.

Разработан алгоритм синтеза оптимальных двухуровневых структур данных.

Разработан алгоритм генерации описания двухуровневой структуры. Согласно данному алгоритму, генерация выполняется за два этапа. На первом этапе генерируются описания всех вспомогательных одноуровневых структур, входящих в состав двухуровневой. На втором этапе генерируется класс самой двухуровневой структуры данных.

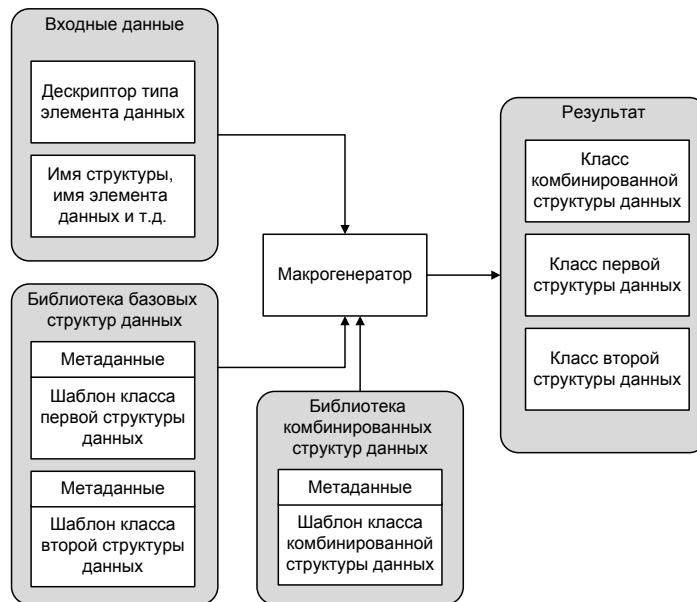


Рис. 4. Генерация описания комбинированной одноуровневой структуры данных

В четвертой главе «Экспериментальные исследования полученных результатов» описаны назначение, состав и функции системы синтеза оптимальных структур данных (см. рис. 5). Программное обеспечение написано на языках *Delphi Pascal* и *C#* и имеет объем ~ 15000 строк.



Рис. 5. Структура системы синтеза оптимальных структур данных

Приведены результаты эксплуатации системы и экспериментальных исследований. Сформулированы цели экспериментальных исследований, даны характеристики объектов.

Целями исследования зависимости вычислительной сложности реализации алгоритма уравновешенной двоичной свертки от структур данных являлись анализ работоспособности модуля синтеза оптимальных двухуровневых структур данных и оценка качества генерируемых структур данных.

В качестве объекта исследования был выбран алгоритм уравновешенной двоичной свертки, который применяется для решения задачи декомпозиции функциональных схем устройств вычислительной техники.

Экспериментальное исследование показало, что применение оптимальных структур данных по сравнению выбранными вручную простыми структурами данных может более чем на порядок сократить время решения задачи. Также происходит значительное сокращение времени, затрачиваемого на разработку структур данных, так как в случае синтеза структуры данных генерируются автоматически.

Целью следующего исследования являлся анализ вычислительной сложности алгоритма при различных наборах входных данных. В качестве объекта исследования был использован алгоритм неуравновешенной двоичной свертки, который применяется для решения задачи декомпозиции функциональных схем устройств вычислительной техники.

По результатам исследования показано, что структуры данных, оптимальные для одного набора входных данных могут быть неоптимальными для другого. Для выбранного алгоритма и заданного ограничения на размер входа задачи, это различие несущественно, так как изменение времени решения задачи в данном случае составляет менее 1 секунды, при погрешности измерения 0,04 с.

Установлено, что для других алгоритмов, в которых количество операций разных типов также зависит от свойств входных наборов данных, подобное различие может быть существенным, поэтому при генерации структур данных необходимо учитывать вышеописанные свойства. В том случае, когда заранее определить свойства входных наборов данных затруднительно, можно использовать методы специализации, описанные в первой главе.

Целями исследования зависимости вычислительной сложности алгоритма лингвистического анализа текста от структур данных являлись анализ работоспособности модуля синтеза оптимальных одноуровневых структур данных и определение эффекта от применения оптимальных одноуровневых структур данных.

В качестве объекта исследования был использован алгоритм лингвистического анализа текста, который применяется в продукте Traffic Monitor компании InfoWatch для решения задачи детектирования конфиденциальной информации в заданном электронном тексте. По результатам исследования показано, что использование оптимальных структур данных позволило ускорить работу программы в среднем на 14%, при этом наиболее заметно снизилось время обработки больших текстов – на 44%.

Показано, что реализация разработанных в теоретической части методов и алгоритмов в виде единой программной системы обеспечивает автоматизацию разработки алгоритмов решения задач структурного анализа и синтеза, что позволяет существенно сократить трудоемкость подготовки таких задач к решению и обеспечить удовлетворяющие разработчика характеристики получаемого программного обеспечения (емкостную и вычислительную сложность).

ОСНОВНЫЕ РЕЗУЛЬТАТЫ РАБОТЫ

1. Для представления простых структур разработаны модели базовых структур данных, позволяющие оценивать вычислительную сложность базовых операций и емкостную сложность исследуемой структуры, а также обладающие меньшей избыточностью по сравнению с моделями, разработанными для языка АЛГОРИТМ.

2. Поскольку в алгоритмах решения задач на графах используются также многоуровневые структуры данных, были определены и сформулированы правила перехода от объекта к модели для многоуровневых и комбинированных структур данных, а также определены расчетные соотношения для оценки вычислительной и емкостной сложности таких структур по построенным моделям.

3. Для получения комбинированных структур данных из базовых структур, разработана и формально определена операция объединения базовых структур данных. Для упрощения внесения изменений в библиотеку шаблонов предложен вариант реализации операции объединения базовых структур, который существенно упрощает как модификацию существующих шаблонов базовых структур данных, так и добавление новых шаблонов структур данных.

4. Выполнена формальная постановка задачи синтеза оптимальных одноуровневых и многоуровневых комбинированных структур данных, результатом решения которой является структура данных, отвечающая требованиям минимальной вычислительной и емкостной сложности для данного набора операций.

5. Для решения задачи синтеза оптимальной структуры данных предложены алгоритмы синтеза одноуровневых и многоуровневых структур, позволяющие получать результирующие структуры посредством комбинации базовых, а также автоматически получать оценки вычислительной и емкостной сложности.

Теоретические результаты работы доведены до практической реализации в виде системы синтеза оптимальных структур данных для алгоритмов решения комбинаторных задач на графах. Экспериментальная проверка и эксплуатация системы подтвердили результаты теоретических исследований и основные научные положения, а также показали эффективность предло-

женных моделей, способов и средств. Так, практическое применение системы при создании программно-аппаратного комплекса БКФ 2.00 в компании InfoWatch позволило сократить время обработки больших электронных текстов на 44% и уменьшить стоимость аппаратной составляющей комплекса на 10%. Также применение оптимальных структур данных при разработке очередной версии системы защиты Protection Builder 5.70 в компании «Протекшен Технолоджи» позволило увеличить скорость защиты приложений в среднем на 20%.

Работы по теме диссертации

1. Иванова Г.С, Пасечников К.А. Макрогенерация описаний структур данных для системы проектирования алгоритмов решения задач структурного синтеза // Современные информационные технологии: Сб. трудов каф., посвященный 175-летию МГТУ им. Н.Э. Баумана. – М.: Эликс+, 2004. – С. 69-73.
2. Пасечников К.А. Анализ проблемы синтеза оптимальных структур данных для алгоритмов решения комбинаторных задач на графах. // Информатика и системы управления в XXI веке: Сб. трудов молодых учёных, аспирантов и студентов МГТУ им. Н.Э. Баумана. – 2007. – №5. – С. 118-124.
3. Пасечников К.А., Иванова Г.С. Модели структур данных для представления объектов задач структурного анализа и синтеза. // Информатика и системы управления: Сб. трудов молодых учёных, аспирантов и студентов МГТУ им. Баумана. – 2006. – №4. – С. 45-48.
4. Пасечников К.А., Иванова Г.С. Генерация оптимальных структур данных для алгоритмов решения комбинаторных задач на графах. // Аэрокосмические технологии: Труды Всероссийских и Международной научно-технических конференций. – М., 2008. – С. 112-115.
5. Пасечников К.А. Генерация комбинированных структур данных // Наука и образование. Инженерное образование: Эл. науч. издание. - 2008. - № 11. (Номер гос. регистрации 0420900025)
6. Пасечников К.А. Модели структур данных с векторной, списковой и древовидной организацией элементов // Наука и образование. Инженерное образование: Эл. науч. издание. - 2008. - № 10. (Номер гос. регистрации 0420900025)
7. Пасечников К.А., Иванова Г.С. Синтез оптимальных структур данных для решения задач на графах // Вестник Московского государственного технического университета им. Н.Э. Баумана. Приборостроение. – 2008. – № 4(73). – С. 29-38.

Подписано к печати

Заказ

Объем 1 п.л.

Тираж 100 экз. Типография МГТУ. 105005, Москва, 2-я Бауманская, 5.