

ПОСТРОЕНИЕ КРИТЕРИЕВ ОЦЕНКИ СТРУКТУРНОЙ СЛОЖНОСТИ В РЕЖИМЕ АДАПТАЦИИ К ЦЕЛЯМ АНАЛИЗА СИСТЕМ БОЛЬШОЙ РАЗМЕРНОСТИ

С.С. Толстых, Р.В. Федоров, В.Е. Подольский

ГОУ ВПО «Тамбовский государственный технический университет», г. Тамбов

Рецензент В.Ф. Калинин

Ключевые слова и фразы: компьютерные сети; образование; сложность; структура; структурная сложность.

Аннотация: Обсуждаются проблемы построения критериев структурной сложности, адаптированных к целям анализа технических систем большой размерности.

В последние годы все чаще появляются задачи, связанные с необходимостью производить оценки структурной сложности систем большой размерности. Региональная образовательная компьютерная сеть (РОКС) является примером системы большой размерности, структурные свойства которой представляют научно-практический интерес и являются предметом исследования [1]. Данная статья является продолжением научных исследований, отраженных в ряде предшествующих публикаций.

Под структурной сложностью системы понимается количественная характеристика качественных представлений о наискорейшем познании этой системы на основе использования математических моделей узлов и весовых характеристик связей между ними с последующим расчетом на вычислительном кластере. Познание структурной сложности системы осуществляется проецированием структурных свойств системы в вещественную скалярную величину – структурную сложность. Для ее оценки необходим соответствующий критерий, ассоциирующий цели анализа, структурные свойства и вычислительные аспекты исследуемой системы (или класса систем).

Ранее были предложены рекурсивный [1] и матричный [2] критерии структурной сложности (КСС). В первом случае структура системы определяется взвешенным орграфом со структурной сложностью, оцениваемой значением критерия $S(G)$, посредством которого орграфу G в однознач-

Толстых С.С. – кандидат технических наук, доцент кафедры «Информационные процессы и управление» ТамбГТУ; Федоров Р.В – аспирант кафедры «Информационные процессы и управление» ТамбГТУ; Подольский В.Е – доктор технических наук, профессор, проректор по информатизации ТамбГТУ, г. Тамбов.

ное соответствие ставится вещественное положительное число $S(G) \in R^+$, причем, чем больше значение $S(G)$, тем граф G считается более сложным. Таким образом, матричный критерий позволяет построить шкалу структурной сложности для попарного сравнения орграфов как абстрактных математических объектов.

Для анализа структурных свойств *технических* систем большой размерности в большей степени подходит рекурсивный критерий, архитектура построения которого подразумевает подчиненность целям анализа системы и учет морфологии системы.

Дальнейшим развитием теории структурной сложности является построение критериев оценки структурной сложности в автоматизированном режиме – необходимо задать цели анализа, аксиоматику сложности применительно к классу рассматриваемых систем и вывести конкретный вид критерия как решение системы операторных уравнений.

Рассмотрим в общем виде рекурсивный критерий структурной сложности

$$\Theta(\Pi) = \begin{cases} |Z| = 0 \Rightarrow \square; \\ |Z| = 1 \Rightarrow \min_{1 \leq k \leq v(v)} (\bar{\gamma}_k \otimes (\square \oplus \Theta(\bar{\Pi} \setminus \{\bar{\pi}_k\}))) ; \\ |Z| > 1 \Rightarrow \sum_{i=1}^{|Z|} \Theta(Z_i), \end{cases} \quad (1)$$

где Π – множество дуг системы; $\Theta(\Pi)$ – структурная сложность; Z – множество подсистем, образовавшихся как результат структурной декомпозиции, $Z = \text{Dec}(\Pi)$; $\text{Dec}(\Pi)$ – оператор структурной декомпозиции, он выявляет $|Z|$ сильносвязанных подсистем; \square, \square – соответственно, нулевая и единичная структурные сложности; \oplus, \otimes – операции сложения и умножения сложностей соответственно; $\bar{\gamma}_k$ – вес дуги $\bar{\pi}_k$; $\bar{\Pi}$ – множество дуг системы после структурной минимизации $\bar{\Pi} = \text{Gm}(\Pi)$; $\text{Gm}(\Pi)$ – оператор структурной минимизации; $\bar{\Pi} \setminus \{\bar{\pi}_k\}$ – множество дуг системы без дуги $\bar{\pi}_k$.

В формуле (1) некоторые ее части могут адаптироваться к целям анализа, в частности:

- 1) оператор минимизации Gm ;
- 2) оператор декомпозиции Dec ;
- 3) алгоритм выбора дуги $\bar{\pi}_k$;
- 4) алгоритм вычисления сложности исходной системы при удалении дуги $\bar{\pi}_k$, исходя из сложности остаточной системы и весовых характеристик дуги $\bar{\pi}_k$;

5) весовые характеристики дуг могут быть фреймами знаний, что требует формализации не только самих фреймов знаний, но и аксиоматики сложения и умножения, нулевой и единичной сложности.

Существующая библиотека PLib, которая является базовым инструментальным средством исследования структурной сложности замкнутых, статических, детерминированных систем большой размерности, позволяет изменять только тип весовых характеристик дуг, так как изначально была разработана с учетом возможности использования фреймов знаний в качестве весовых характеристик дуг.

При разработке новых КСС изменения остальных частей алгоритма требуют немалых усилий. Для преодоления обозначенных сложностей было решено доработать библиотеку PLib, что послужило началом работ по проектированию и созданию дополнительного программного модуля, позволяющего осуществлять построение КСС, адаптированных под цели анализа.

В качестве языка для задания спецификаций КСС был выбран язык XML. Этому способствовали универсальность языка XML и развитые средства работы с данными в формате XML – XML DOM, XSLT, XPath.

Формализацию участков алгоритма КСС решено осуществлять с использованием выражений состоящих из:

- переменных;
- констант;
- арифметических операторов (сложение, умножение);
- логических операторов («и», «или», «=», «>», «<»);
- квантификаторов существования « \exists » и всеобщности « \forall ».

Синтаксически выражения представляют собой XML-код. Семантически – это древовидные структуры, состоящие из узлов следующих типов: переменные, константы, операторы. Каждый дочерний узел связан с родительским определенным типом связи: операнд, условие (для условий квантификаторов).

Критерий структурной сложности в общем виде должен состоять из следующих частей:

- константной;
- изменяемой;
- подключаемых модулей.

Константная часть – базис (каркас) алгоритма вычисления критерия структурной сложности. Изменяемая часть – участки критерия, доступные для адаптации под цели анализа структурной сложности. Изменяемая часть критерия задается пользователем в виде алгебраических спецификаций. Подключаемые модули предназначены для дополнения возможностей алгебраических спецификаций. Примером подключаемых модулей могут служить модули операторов, используемых в алгебраических спецификациях изменяемой части критерия. Функционал подключаемых операторов достаточно сложен для его формализации с помощью алгебраических спецификаций и последующего построения их программного кода.

На рис. 1 представлена DFD-диаграмма верхнего уровня, отражающая процесс построения КСС.

Пользователь формирует спецификацию критерия в формате XML. В спецификации заданы адаптируемые участки алгоритма расчета структурной сложности. Например:

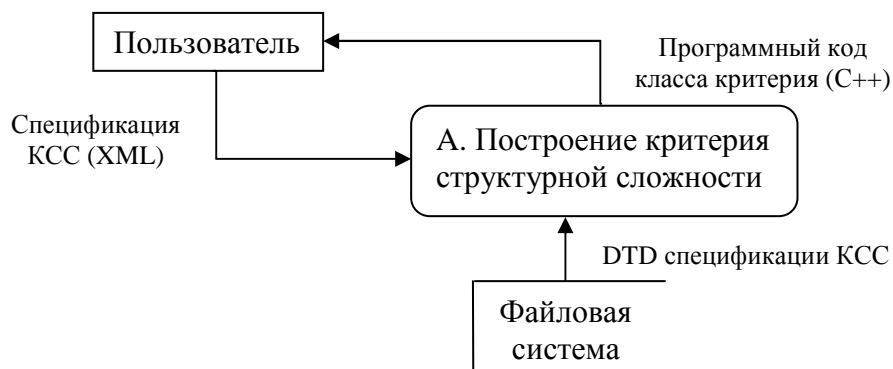


Рис. 1. Построение критерия структурной сложности

- оператор структурной минимизации;
- оператор структурной декомпозиции;
- определены фреймы знаний, характеризующие вес дуг;
- определена аксиоматика сложения и умножения сложностей, нулевой и единичной сложности.

Спецификация проверяется согласно DTD. После чего происходит построение программного кода критерия.

Рассмотрим детализацию процесса А. «Построение критерия структурной сложности» (рис. 2).

При построении программного кода класса критерия, для каждого участка критерия, возникает необходимость построения программного кода, реализующего алгоритм участка на основе его алгебраической спецификации (в формате XML). Для этого создается объект алгебраической спецификации на основе данных XML-спецификации (процесс «А.3») и происходит дальнейшая его передача процессу построения программного кода участка критерия (процесс «А.4»).

Процессы «А.3» и «А.4» выполняются до тех пор, пока главный процесс «А.2» не завершит чтение спецификации КСС и не построит программный код всего класса критерия.

На основании спецификации каждого участка алгоритма осуществляется построение программного кода КСС. Весь программный код компонуется в единый класс на языке C++. Класс является потомком абстрактного класса критерия и реализует виртуальные методы, определенные в абстрактном классе.

В ходе работ по проектированию модуля построения КСС в качестве базисных шаблонов проектирования было решено использовать следующие:

- шаблон «Composite» предназначен для объединения объектов в древовидные структуры, дает клиентам единообразный способ работы с индивидуальными объектами и их объединениями;
- шаблон «Generation Gap» предназначен для облегчения модификации сгенерированного кода.



Рис. 2. Детализация процесса А. «Построение критерия структурной сложности»

Для реализации модуля построения КСС был выбран язык PHP 5. Основными факторами, сыгравшими роль при выборе языка для реализации библиотеки, являются:

- процесс построения КСС, который неразрывно связан с генерацией программного кода, то есть работой со строковыми данными;
- возможность работы с XML DOM, XPath;
- легкость и скорость разработки;
- возможность работы интерпретатора на самых различных платформах.

Рассмотрим пример задания оператора структурной минимизации.

Пусть есть система ∞ , заданная на объектах формальных знаний $F = \{f_i, i = 1, \dots, n\}$, конкретизируемая дугами Π , которые, в свою очередь, конкретизируются весами $\Gamma: \Sigma \mathcal{M}((\Pi \mathcal{M} \Gamma) \supseteq F)$. Рассмотрим пример системы операторных уравнений, решение которой является оператором структурной минимизации графа G_m . Структурная минимизация графа – удаление транзитивных вершин в графе. Пример минимизации показан на рис. 3.

Смысл минимизации состоит в стягивании дуг, проходящих через вершины f_1, f_2, f_3, f_4 , и образовании обобщенной дуги из f_1 в f_4 . Перечислим формулы необходимые для минимизации:

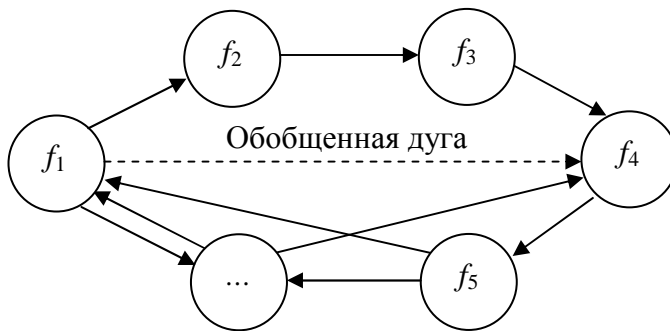


Рис. 3. Минимизация

1) формула минимизации, определяющая множество транзитивных вершин, подлежащих минимизации; минимизация проходит последовательно, пока есть такие множества;

2) формула, определяющая правило назначения весов при стягивании вершин;

3) формула, определяющая правило изменения объектов формальных знаний при стягивании вершин;

4) формула, определяющая правило переназначения веса уже существующей дуги при схлопывании с дугой, образовавшейся в результате минимизации;

5) формула, определяющая правило изменения объектов формальных знаний при схлопывании дуг;

6) общая формула, объединяющая пять предыдущих в единое целое.

Рассмотрим формулу минимизации. Основной задачей, решаемой с помощью этой формулы, является поиск транзитивных дуг. Корневым оператором в формуле минимизации является квантификатор существования, так как это единственный оператор, позволяющий неявно осуществлять поиск значений индексов дуг.

Рассмотрим формулу для поиска транзитивных дуг,

$$\begin{aligned}
 \text{Gmb}(\Sigma) : (\Sigma \mathfrak{M}((\Pi \mathfrak{M} \Gamma) \supseteq F)) &\rightarrow \{f_i, f^*, f_j\} \mid \\
 (\exists ((\pi_i, \pi_j) = ((f_i \mapsto f^*), (f^* \mapsto f_j)) \in \Pi, \gamma_i \& \text{de}(\pi_i), \gamma_j \& \text{de}(\pi_j)) \mathfrak{M} & \quad (2) \\
 \mathfrak{M}(\exists f_l, l \in \{1, \dots, n\}: (((f_l \mapsto f^*) \theta \supseteq (l \neq i)) \& ((f^* \mapsto f_l) \supseteq (l \neq j))) \in \Pi)) &
 \end{aligned}$$

Существуют две дуги – π_i из f_i в f^* с весом γ_i и π_j из f^* в f_j с весом γ_j , такие, что:

– не найдется ни одна другая дуга, которая входила бы в их общую вершину f^* , отличная от π_i ;

– не найдется ни одна другая дуга, которая выходила бы из их общей вершины f^* , отличная от π_j .

Корневой оператор в формуле (2) – квантификатор существования двух дуг $f_i \mapsto f^*$ и $f^* \mapsto f_j$. Обе дуги должны иметь общую вершину f^* .

С алгоритмической точки зрения, квантификатор существования реализуется как перебор по трем переменным в трех вложенных циклах. Если в результате этого перебора подтверждается гипотеза о существовании пары дуг, соответствующих условию в теле квантификатора – циклы прерываются, и становятся известны индексы 3-х вершин, через которые проходит искомая пара дуг.

Вложенный квантификатор существования осуществляет итерации по одной переменной, так как значение другой (индекс вершины f^*) считается известным – переменная участвует в переборе во внешних циклах.

После того как транзитивные вершины найдены, начинается собственно процесс минимизации. К этому моменту предполагается, что пара дуг, участвующих в минимизации, уже найдена.

Рассмотрим процесс стягивания вершин – главный этап минимизации. Как видно из рис. 4, после стягивания дуг с весами γ_x, γ_y образуется дуга с весом γ_c :

$$\text{Gmc}_\gamma: (\gamma_x, \gamma_y) \rightarrow \gamma_c \mid \gamma_c \& \min(\gamma_x, \gamma_y). \quad (3)$$

Помимо создания новой дуги при стягивании вершин должно происходить изменение состояния одного из объектов формальных знаний (f_x или f_y):

$$\text{Gmc}_f: (f_x, f_y) \rightarrow f \mid f \& (f_x, f_y). \quad (4)$$

Рассмотрим схлопывание дуг – побочный эффект стягивания дуг. Как видно из рис. 5, после стягивания дуг с весами γ_x и γ_y образовалась дуга с весом γ_c . Но в графе уже существует дуга, проходящая через те же самые вершины. Зададим операторное уравнение, определяющее формулу изменения веса существующей дуги

$$\text{Gmo}_\gamma: (\gamma_z, \gamma_c) \rightarrow \gamma_o \mid \gamma_o \& \gamma_z + \gamma_c; \quad (5)$$

и изменение состояния объектов формальных знаний (f_x и f_y):

$$\text{Gmo}_f: (f_x, f_y) \rightarrow f \mid f \& f_x \cap f_y. \quad (6)$$

Дополнив уравнения (2)–(6) общим уравнением оператора минимизации, получим систему операторных уравнений, решение которой и будет являться искомым оператором структурной минимизации. Общее уравне-

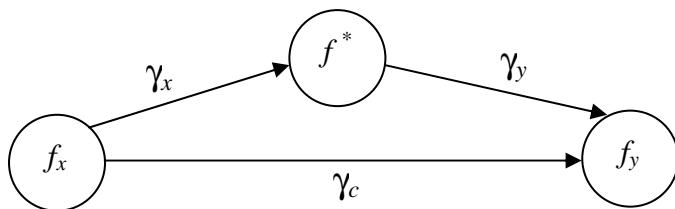


Рис. 4. Минимизация – стягивание дуг

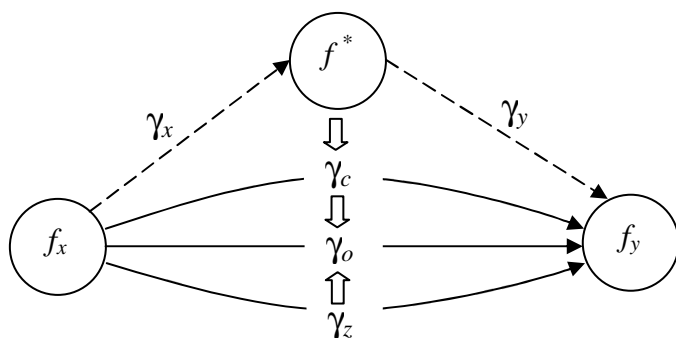


Рис. 5. Минимизация – схлопывание дуг

ние оператора минимизации является связующим для уравнений (2)–(6). Оно определяет последовательность выполнения операций стягивания и схлопывания дуг и обратимость оператора.

Построение критериев оценки структурной сложности в автоматизированном режиме, является очень важным этапом развития теории структурной сложности. Это позволит в дальнейшем создавать КСС, адаптированные к целям анализа, сосредоточившись на задачах исследования интересующей нас технической системы, а не на изучении интерфейсов расчетных модулей и реализации требуемого критерия структурной сложности.

Список литературы

1. Подольский, В.Е. Повышение эффективности региональных образовательных компьютерных сетей с использованием элементов структурного анализа и теории сложности / В.Е. Подольский, С.С. Толстых. – М. : Машиностроение, 2006. – 176 с.
2. Толстых, С.С. Матричный критерий структурной сложности замкнутых систем / С.С. Толстых // Вестн. Тамб. гос. техн. ун-та. – 1998. – Т.4, № 2–3. – С. 238–244.

Building of Structural Complexity Estimating Criteria in Adaptation Mode for Analysis Objectives of Systems with Large Dimension

S.S. Tolstykh, R.V. Fedorov, V.E. Podolsky

Tambov State Technical University, Tambov

Key words and phrases: computer networks; education; complexity; structure; structural complexity.

Abstract: The problems of building of structural complexity estimating criteria adapted for analysis objectives of technical systems with large dimension are discussed.

© С.С. Толстых, Р.В. Федоров, В.Е. Подольский, 2008