

М. Ю. Пименов

Обоснование состава дефектов сложных электротехнических комплексов

Научный руководитель: проф. В. С. Дрогайцев

Аннотация. Предлагается подход к обоснованию состава возможных дефектов в структуре сложного электротехнического комплекса. Главной особенностью подхода является использование процедуры декомпозиции исходной модели технического объекта, по результатам которой происходит переход от компонентов системы к агрегатам компонентов. Формирование агрегатов осуществляется исходя из функциональной значимости компонентов. Декомпозиция исходной системы необходима для минимизации пространства дефектов, из которого исключаются неисправности малозначимых компонентов.

Ключевые слова и фразы: декомпозиция модели, обоснование неисправностей, алгоритм.

1. Введение

Разработка системы диагностирования класса технических объектов осуществляется таким образом, чтобы программными и аппаратными средствами поддерживать методы и алгоритмы обнаружения дефектов, основанные на анализе поведения диагностируемого объекта. Так, задача диагностирования состояния объекта заключается в сопоставлении значений параметров, зафиксированных в системе, с шаблонными комбинациями, отождествлёнными с известным подмножеством дефектов. Синтез причинно-следственных связей между кодовыми комбинациями параметров и кодами дефектов происходит на стадии моделирования функциональных зависимостей объекта диагностирования (ОД) с привлечением данных и знаний, зафиксированных в техническом задании и технических условиях на объект. Процедура моделирования рассматривает допустимое подмножество дефектов, состав которого должен быть обоснован исходя из заданных качеств потенциально неисправных компонентов системы.

С целью обоснования допустимого подмножества дефектов предлагается синтезировать модель ОД [1, 2], подвергнуть её процедуре декомпозиции для выделения элементов декомпозиции — агрегатов.

Элементы декомпозиции однозначно определяют состав допустимых дефектов объекта.

2. Формализм модели объекта диагностирования

В качестве средства описания ОД используется гиперграфовая модель [2].

Гиперграф, описывающий структурно-конструктивные, функциональные, информационные и временные особенности сложных электротехнических комплексов есть кортеж

$$(1) \quad H(V, E, \gamma, \theta, \alpha, \omega) \in \Theta,$$

где Θ — множество гиперграфовых структур для реализации иерархического подхода к описанию ОД; V — конечное множество вершин, каждая из которых отождествляется с компонентом или агрегатом компонентов (гиперграфом); E — конечное множество гиперребер, характеризующих пути распространения сигналов, отождествлённые с каналами управления; γ — отображение, связывающее вершину и гиперребро с явным указанием индекса вершины в ребре; θ — показатель того, является ли вершина самостоятельным гиперграфом; α — отображение элементов гиперграфа на множество атрибутов; ω — отображение элементов, характеризующихся атрибутами, на множество значений атрибутов.

Гиперрёбра представляют собой упорядоченный вектор вершин (компонентов или агрегатов). Упорядочение производится по пути распространения сигналов внутри целевого объекта, то есть гиперрёбра представляют собой явные каналы управления:

$$(2) \quad \gamma : V \times E \rightarrow r,$$

где $r = \{1, 2, \dots, N(E)\}$ натуральное число, указывающее на индекс вершины $v \in V$ в ребре $e \in E$. В таком случае вершина v_i имеет индекс a в гиперребре e_j , если $\gamma(v_i, e_j) = a$.

Показателем того, является ли вершина самостоятельной подструктурой, служит отображение

$$(3) \quad \theta : V \rightarrow \Theta,$$

которое представляет собой новый гиперграф $\exists H_i : \theta(v_i) \in \Theta, H_i = \theta(v_i)$, отождествлённый с вершиной v_i . Для вершины, не являющейся отдельным гиперграфом, вводится значение *null*:

$$(4) \quad \exists v_i : \theta(v_i) = null.$$

Отображение

$$(5) \quad \alpha : V \cup E \rightarrow M,$$

где M — некоторое семейство непустых подмножеств множества атрибутов A , является надстройкой для получения помеченного гиперграфа.

Отображение

$$(6) \quad \omega : (V \cup E) \times A \rightarrow R$$

каждой паре значений (элементу и его атрибуту) ставит в соответствие значение атрибута. R — множество значений атрибутов.

Таким образом, элемент c гиперграфа H имеет атрибут a_i со значением b_j , если $a_i \in a(c)$ и $b_j = \omega(c, a_i)$.

Структурно-конструктивная модель ОД заключена в паре $\langle V, E \rangle$, в то время как функциональные, информационные и временные аспекты модели содержатся в атрибутах множества A и реализуются с помощью отображений $\langle \alpha, \omega \rangle$.

Компоненты системы характеризуются функциональной значимостью, которая оценивается на стадии проектирования объекта, нормируется и обосновывается в технической документации. Показатель функциональной значимости определяет роль компонента как обособленной единицы в сложной информационной системе, то есть его функциональные возможности. Данные о значимости компонентов системы содержатся в модели объекта. Пространство атрибутов дополняется атрибутом функциональной значимости, а область значений атрибутов — значениями показателя значимости на каждый компонент системы:

$$(7) \quad a_{FV} \in A, \forall v_i \in V : \alpha(v_i) = a_{FV}, \omega(v_i, a_{FV}) = b_k, b_k \in R.$$

Для связи гиперрёбер с вершинами служит матрица инциденции. Матрица инциденции гиперграфа $H(V, E, \gamma, \theta, \alpha, \omega)$ формируется руководствуясь правилом:

$$(8) \quad A(H) = \|a_{ij}\|, a_{ij} = \begin{cases} 1, & \text{если } v_i \in e_j; \\ 0, & \text{если } v_i \notin e_j. \end{cases}$$

3. Постановка задачи декомпозиции

Задача декомпозиции заключается в распределении множества вершин V гиперграфа по k подмножествам gr_1, \dots, gr_k таким образом, чтобы удовлетворялся набор декомпозиционных ограничений, а оптимизируемая функция принимала экстремальное значение. Множества gr_1, \dots, gr_k попарно не пересекаются и при объединении составляют исходное множество V :

$$(9) \quad gr_i \cap gr_j = \emptyset, i, j = \overline{1, k}, i \neq j, \sum_{i=1}^k gr_i = V.$$

Значение k , то есть число групп компонентов декомпозиции, неизвестно. Такая постановка задачи отлична от изложенной в [2], и позволяет управлять процессом разбиения структуры (числом элементов декомпозиции) варьируя параметры критерия и ограничений метода.

В основе ограничений и критерия лежит использование весовых характеристик элементов гиперграфа.

Веса вершин заключены в атрибутах

$$(10) \quad a_{FV} \in A, \omega(v_i, a_{FV}) = b_k, b_k \in R$$

и известны до начала процесса декомпозиции:

$$(11) \quad \forall v_i \in V : w_i = \omega(v_i, a_{FV}).$$

Процедура декомпозиции исходной структуры гиперграфа, представляющего модель ОД, фактически реализует следующие задачи:

- для нулевого уровня иерархии, представленного вершинами, которые являются функциональными узлами исходной структурно-функциональной схемы, производится декомпозиция, результатом которой является отнесение каждой вершины графа к тому или иному классу;
- для данного уровня иерархии агрегирование компонентов в гиперграфы, наследующие свойства составных элементов;
- постановка и ход декомпозиции исходя из результатов декомпозиции системы на предыдущем этапе (уровне иерархии).

Переход от групп компонентов к новой структуре гиперграфа следующего уровня иерархии включает следующие действия: каждая группа представляется новым гиперграфом — подграфом разбиения для нижнего уровня иерархии; в качестве исходных данных

для продолжения процесса декомпозиции с целью выделения новых уровней иерархии и состава актуальных им агрегатов используется новый сформированный гиперграф, в состав которого входят подграфы разбиения, полученные на текущем этапе декомпозиции.

Группе компонентов gr_i ставится в соответствие новый подграф H_i , который связывается с новой вершиной v_k . При этом исходный граф H изменяется и принимает вид H' :

$$(12) \quad \begin{aligned} & \forall gr_i \in Gr, !H_i(V_i, E_i, \gamma_i, \theta_i, \alpha_i, \omega_i) : \\ & V_i = \{v_j \in gr_i\}, E_i = \{e^* : \exists e \in E : e^* \subseteq gr_i, e \cap e^* \neq \emptyset\}, \\ & !v_k : \theta(v_k) = H_i, \\ & V' = V \setminus gr_i \cup \{v_k\}, E' = \{e^* : \exists e \in E : e^* = e \setminus gr_i\}. \end{aligned}$$

В результате происходит стягивание групп вершин в новую вершину v_k .

4. Разработка метода декомпозиции

Ограничение метода декомпозиции имеет вид:

$$(13) \quad \begin{aligned} w(gr_i) &= \sum_{v_q \in gr_i} w(v_q), \\ Low &\leq w(gr_i) \leq Up, \quad i = \overline{1, k}, \end{aligned}$$

где Low, Up — нижняя и верхняя границы отрезка, внутри которого распределяются веса групп вершин. Из ограничения следует, что веса групп вершин усредняются к некоторому среднему значению, находящегося в отрезке $[Low; Up]$:

$$(14) \quad \bar{w} = (Low + Up)/2.$$

Критерием качества декомпозиции является минимум ошибки полученных весов агрегатов вершин к среднему значению:

$$(15) \quad K = \sum_{gr_i \in Gr} (w(gr_i) - \bar{w})^2 \rightarrow min.$$

Минимум критерия K определяется, с одной стороны, конструктивными особенностями рассматриваемого объекта (топологией гиперграфа), а с другой, видом распределения значений показателя значимости компонентов (весов вершин). Алгоритм декомпозиции гиперграфа, учитывающий пороговую характеристику весов агрегатов и взаимную смежность гиперрёбер, состоит из следующих шагов.

- (1) Гиперрёбра сортируются в порядке невозрастания веса или в случае совпадения весов — в порядке неубывания размера

$$(16) \quad E^{(1)} = \{e_i : w(e_i) > w(e_{i+1}) \mid N(e_i) < N(e_{i+1}) \\ \& w(e_i) = w(e_{i+1})\},$$

где $N(\cdot)$ — здесь и далее оператор размера произвольного множества.

- (2) Для первого гиперребра определяется состав смежных ему гиперрёбер, который сортируется в порядке невозрастания размера гиперрёбер. Для этого реализуется отображение

$$(17) \quad \begin{aligned} & INC : E \rightarrow E, \\ & \forall e_i \in E^{(1)} : INC(e_i) = \{e_j : N(e_i \cup e_j) > 1\}. \end{aligned}$$

- (3) Состав гиперрёбер изменяется таким образом, чтобы исключить наличие смежных гиперрёбер

$$(18) \quad \forall e_i \in E^{(1)} : E^{(2)} = \{e_j : e_j = e_i \setminus e_k\} \cap \{e_k : \forall e_k \in INC(e_i)\}.$$

- (4) Повторяются п.1–3 до тех пор, пока не останется ни одной пары смежных гиперрёбер.

- (5) Производится разделение гиперрёбер на два новых ребра, если его вес превышает верхнюю границу ограничения Up

$$(19) \quad \begin{aligned} & \forall e_i \in E^{(2)} : \\ & E^{(3)} = \begin{cases} \{e_i\}, & \text{если } w(e_i) < Up; \\ \{e_j, e_k\} & : e_j, e_k \subset e_i; e_j \cup e_k = e_i; N(e_j \cap e_k) = 1. \end{cases} \end{aligned}$$

- (6) Производится объединение рассматриваемого гиперребра с соседним в случае, если его вес меньше нижней границы Low

$$(20) \quad \forall e_i \in E^{(3)} :$$

$$E^{(4)} = \begin{cases} \{e_i\}, & \text{если } w(e_i) > Low; \\ \{e_i^* : (\bigcup_{v_j \in e_i} v_j) \cup (\bigcup_{v_r \in e_k} v_r) : \exists e_k : N(e_i \cap e_k) = 1\}. \end{cases}$$

- (7) Рёбра упорядочиваются в порядке невозрастания веса или неубывания размера в случае равенства весов

$$(21) \quad E^{(5)} = \{e_i : w(e_i) > w(e_{i+1}) \mid N(e_i) < N(e_{i+1}) \\ \& w(e_i) = w(e_{i+1})\}.$$

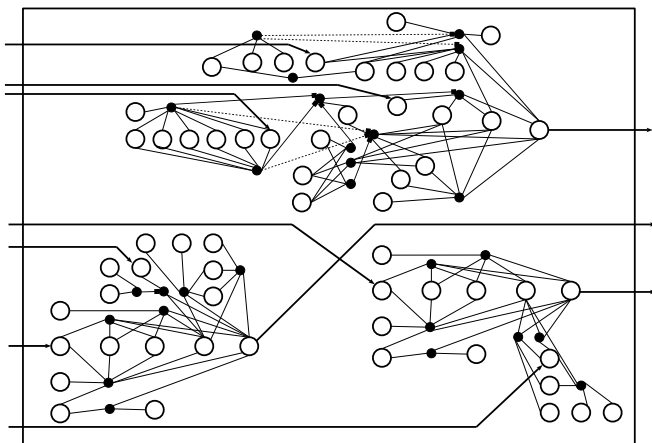


Рис. 1. Исходная структура гиперграфа в Кёниговом представлении

- (8) Для каждого гиперребра все вершины переносятся в отдельную группу-агрегат

$$(22) \quad \forall e_j \in E^{(5)} : Gr = \{gr_i : \forall gr_k \in Gr : gr_i = e_j, gr_i \cap gr_k = \emptyset\}.$$

В результате выполнения алгоритма формируются группы вершин, каждая из которых отождествлена с компонентом рассматриваемой системы.

По результатам декомпозиции на основании полученного гиперграфа $H^m(V^m, E^m, \gamma^m, \theta^m, \alpha^m, \omega^m)$ обосновывается состав допустимых дефектов: каждая вершина множества V^m , представляющая собой подструктуру разбиения, рассматривается как новый компонент, который может быть либо исправен, либо неисправен.

5. Пример декомпозиции гиперграфовой структуры

Руководствуясь изложенным выше методом декомпозиции структуры гиперграфа, осуществлён численный эксперимент по выделению элементов декомпозиции. Исходная структура гиперграфа приведена на рис. 1. В её составе 59 вершин и 30 гиперрёбер. Веса гиперрёбер заключены в отрезке [2; 13].

Декомпозиционные ограничения характеризуются парой $Low = 2$, $Up = 6$.

По результатам декомпозиции гиперграфа получено 22 элемента разбиения, веса которых заключены в отрезке $[2; 5, 1]$, средний вес агрегата равен 3,5. Среднее квадратичное отклонение составило:

$$(23) \quad S(\overline{w^{agr}}) = K = \sum_{v_i \in V, \exists \theta(v_i)} (w(v_i) - \overline{w^{agr}})^2 = 0,89.$$

6. Заключение

Предлагаемый в настоящей публикации метод декомпозиции модели сложного объекта позволяет перейти к модели системы меньшей размерности. Полученная модель системы включает самостоятельные подструктуры, состав которых обоснован с позиции значимости входящих в них элементов.

Элементы декомпозиции отождествляются с допустимыми дефектами. Таким образом, происходит сужение пространства состояний, изначально соответствующее пространству компонентов исходной модели системы.

Список литературы

- [1] Пименов М. Ю. *Описание структуры электротехнических средств в задаче технической диагностики* // Вестник СГТУ. — 2, № 3 (41), 2009, с. 244–247.
- [2] Батищев Д. И. *Многоуровневая декомпозиция гиперграфовых структур* // Информационные технологии. Приложение, № 5, 2008, с. 2–9.

САРАТОВСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ

M. Y. Pimenov. *Faults justification method for complex electrotechnical systems* // Proceedings of Junior research and development conference of Ailamazyan Pereslavl university. — Pereslavl, 2010. — p. 194–201. (*in Russian*).

ABSTRACT. Approach to justify the possible faults of complex electrotechnical system is proposed. The main feature of approach is to use of decomposition of the initial object model to make a transition from components to aggregates of the components. The process of aggregation is based on estimation functional values of the components. Decomposition of the initial system aims to minimize faults range. The latter forms by removing faults identified with irrelevant components.

Key Words and Phrases: model decomposition, faults justification, algorithm.