

УДК 681.142.001.4

АВТОМАТИЗИРОВАННАЯ СИСТЕМА СИНТЕЗА И АНАЛИЗА ПРОГРАММ КОНТРОЛЯ И ИСПЫТАНИЙ СЛОЖНЫХ УСТРОЙСТВ БОРТОВОЙ АВТОМАТИКИ

© 2014 В.А. Прилепский, А.Н. Коптев Ю.В. Коваленко

Самарский государственный аэрокосмический университет
имени академика С.П. Королева

Поступила в редакцию 09.09.2014

В статье рассматривается научно-техническая проблема синтеза программ контроля и испытаний автоматических устройств сетевых структур бортовых комплексов электрооборудования самолетов с применением методов теории образов для решения производственных задач оценки технического состояния готовых изделий электротехнического производства самолетостроительных предприятий.

Ключевые слова: синтез, анализ, система, программа, контроль, испытания, устройство, бортовое электротехническое оборудование

Актуальность проблемы. В современном производстве летательных аппаратов (ЛА) одной из важнейших проблем является проблема точности обеспечения стабильности параметров, надежности и ресурса работы комплекса бортового оборудования в достаточно жестких условиях в течение длительного времени. Остро стоит эта проблема в отношении электротехнического оборудования (ЭТО), обеспечивающего электропитание всех систем бортового оборудования, их защиту и управление ими, связанного с электрическими соединениями, которые определяют эксплуатационную устойчивость работы этого оборудования. Важно отдавать себе отчет в том, что эксплуатационную устойчивость ЭТО обуславливает технология монтажа и, как следствие, диктуется необходимость анализа и синтеза технологических процессов по параметрам точности и надежности, учета связи точности технологического процесса с точностью выходных параметров электросборок этого оборудования, осуществления технологического контроля и диагностики на всех этапах его производства. Создание новых методов и средств контроля, диагностики и измерения физических параметров технологических процессов и объектов производства становится определяющим в обеспечении эффективного управления качеством

изделий машиностроения в целом и бортового ЭТО в частности.

Одним из важнейших путей интеграции перечисленных выше направлений повышения стабильности параметров, надежности и ресурса работы ЭТО, а также снижения трудоемкости, в частности, контроля и испытаний этого оборудования, является развитие автоматизации, обеспечивающей в современных условиях качественно новые этапы повышения эксплуатационной устойчивости ЭТО за счет передачи функций человеческого труда различным техническим системам, построенных с применением идей искусственного интеллекта, реализуемых как гибкие автоматизированные производственные системы (ГАПС). Большое число работ посвящено отдельным вопросам теории и практики контроля и испытаний ЭТО. К ним следует отнести работы П.П. Пархоменко, А.Г. Евланова, А.Н. Коптева, В.А. Прилепского, Н.А. Соловьева, В.П. Чипулиса, Г.Чжена, Е.Меннига, Г.Метца и др. Отсутствие системных исследований привело к тому, что задачи оптимального синтеза программ контроля и испытаний практически не ставились. В результате сложившегося положения сдерживалась разработка методов и средств автоматизации программирования процессов контроля и испытаний.

В данной работе рассматриваются теоретические основы метода представления ОК, их анализа и синтеза, монтажа, контроля и испытаний объектов и систем ЭТО ЛА, методы и алгоритмы, разработанные для решения задач создания подсистемы автоматизированного проектирования программы контроля (САПР ПК), а

Прилепский Василий Андреевич, кандидат технических наук, доцент кафедры эксплуатации авиационной техники. E-mail: 2pr@inbox.ru

Коптев Анатолий Никитович, доктор технических наук, профессор, заведующий кафедрой эксплуатации авиационной техники. E-mail: eat@ssau.ru

Коваленко Юлия Витальевна, аспирантка

также реализацию предложенных решений в виде подсистем автоматизированных систем проектирования программ контроля и испытаний (АПП КИ ЭТОС).

Цель работы: разработка теоретических основ, методик и алгоритмов автоматизации проектирования контроля и испытаний этого оборудования, обеспечивающих гибкость и динамичность производства. Эффективность и качество монтажно-испытательных работ путем создания в составе ГАПС ЭТО авиационных предприятий, САПР ПК.

Достижение указанной цели связано с решением задачи: разработка теоретических и алгоритмических основ проектирования программ контроля и испытаний объектов ЭТО ЛА. Решение данной задачи охватывает теоретические проблемы искусственного интеллекта, которые будут положены в основу проектирования программ контроля и испытаний. В качестве такой основы будет использована теория образов У. Гренандера [1], в частности, использован один важный класс алгебр изображений в четырехмерном пространстве. Опорное пространство $X = R^3 \times R'$, где R' - пространство времени. Образуемыми программы являются некоторые высказывания, в которых роль индекса α образующей заключается в разделении движений (воздействий) на различные типы, а программа, состоящая из этих образующих G , будет репертуаром движений. Преобразования состояний объекта контроля будут включать в себя сдвиги по времени $t \rightarrow t + \Delta t$. Воздействия на показатели связей образующих будут сводиться к тому, что они примут значения $t_{in} + \Delta t, t_{om} + \Delta t$.

Одной из центральных задач проектирования ПК состояния реального автомата ЭТОС является задачей настолько высокой размерности, что ее невозможно решить стандартными методами специалистами, занимающимися производственным контролем, обладающих знаниями лишь технических вопросов. Все изложенное выше – это нерешенная проблема в области искусственного интеллекта. Решение этой проблемы в данной работе базируется на специальном подходе в рамках исчисления предикатов первого порядка, которое может быть успешно модифицировано под задачу проектирования программ контроля сложных автоматических и автоматизированных устройств бортового электрооборудования путем введения ситуации: известна схема устройства, к которой необходимо применить множество воздействий, чтобы получить ответ на вопрос: каково его текущее состояние? При такой постановке вопроса каждую конкретную задачу придется решать самостоятельно на базе набора эвристик.

В работе предлагается новая стратегия развиваемого в рамках теории исчисления предикатов первого порядка с введением дополнительного функционального аргумента. В частности, если имеется предикат первого порядка, например, двухместный предикат, то в предлагаемом построении описания взаимодействия между переменными мы вводим переменную состояния:

$$P(x, y) \rightarrow P(x, y, s). \quad (1)$$

В общем случае n -местный предикат мы заменяем $(n+1)$ -местным предикатом, оставляя последнее место для переменной состояния, которая может принимать значения (состояния) s_1, s_2, \dots, s_n . Эти состояния связываются в граф посредством различных операторов, например, f_1, f_2 и т.д. (рис. 1).

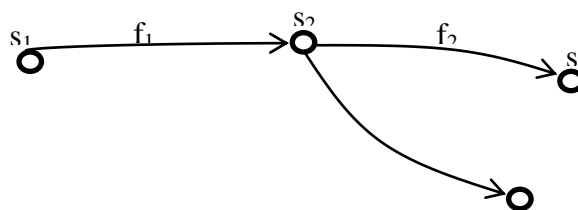


Рис. 1. Граф состояний

Здесь f_1 означает переход из состояния s_1 в s_2 в результате действия оператора f_1 . Такого рода действия могут быть представлены в виде исчисления состояний (образов), поскольку наши операторы отображают состояние на состояние, которые могут быть легко переведены в состояния последовательностной машины или многотактного автомата, представляющего сложную электросборку ЭТО [2, 3]. В силу сказанного, исчисление для построения теории программирования контрольных функций, можно представить как пятерку

$$\Sigma = (P, F, S, \alpha, \Omega), \quad (2)$$

где P – множество предикатов, F – множество операторов, S – множество состояний, α – система аксиом, отображающих перевод из состояния в состояние.

$$P(x, s_1)(F(x, s_1) = s_2) \Rightarrow Q(x, s_2), \quad (3)$$

где $P, Q \in P, F \in F, s_1, s_2 \in S$.

Для произвольного состояния ее можно записать

$$\forall s P(x, s) \Rightarrow Q(x, F(x, s)), \quad (4)$$

где воспользовались подстановкой $F(x, s)$ вместо s_2 , тем самым исключив обозначения двух конкретных состояний. Здесь s соответствует начальному состоянию s_1 , а $F(x, s)$ – конечному состоянию s_2 .

Для описаний начальной ситуации используются схемы аксиом вида

$$H(x, S_H), \quad (5)$$

где $H \in P$, $S_H \in S$, $x \in \Omega$.

$$S_K = F_1(x, F_2(x, \dots, F_{n-1}(x, S_H)), \dots) = F_1, F_2, \dots, F_n(x, S_H). \quad (7)$$

Последовательность функциональных знаков должна пониматься как суперпозиция операторов, которые последовательно применяются к начальному состоянию и переводят его в конечное. Графически наша программа связывает начальные состояния объекта контроля с конечным (рис. 2).

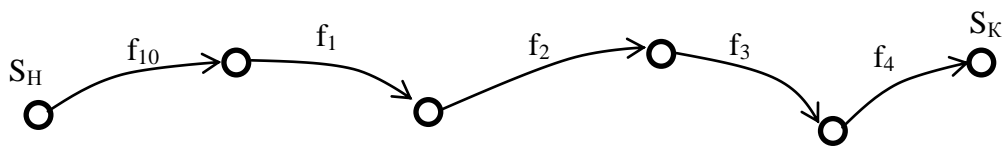


Рис. 2. Графическое представление программы

$$S_K = F_4 \cdot F_3 \cdot F_1 \cdot F_{10}(S_H) \quad (8)$$

Реализация такого подхода требует создания системы, в рамках которой определены предикаты P, F – операторы действий, которые формируют совместно состояния S , а множество объектов, прежде всего включает элементную базу электросборок (базу данных БД), набор операторов, реализуемых системой управления БД (СУБД), математической моделью программы контроля и характеристикой конкретных объектов контроля (таблицы электрических соединений (ТЭС).

В рамках предложенного теоретического подхода формируются принципы автоматизированного проектирования программ контроля электросборок, показана организация входной и выходной информации, кратко представлены некоторые алгоритмы формирования программ контроля электросборок. Одной из важнейших проблем, которая не обсуждается в данном контексте и решаемая в рамках САПР ПК, является проблема размерности. Класс объектов, моделируемых в САПР ПКЭ – электросборки, построенные преимущественно на релейно-контактных и полупроводниковых элементах. САПР ПКЭ является развивающейся системой, поэтому при ее применении необходимо учитывать перечень элементов, обрабатываемых системой: реле, контакторы, диоды, резисторы, конденсаторы,

Теперь мы формулируем задачу контроля в этой записи как достижения конечного состояния, для которого выполняется условие $K(x, S_K)$,

$$(\exists S_K) \{M(x, S_K)\}. \quad (6)$$

Задача получения программы контроля ставится как теорема, которую следует доказать на основе аксиом исчисления. Программа должна иметь следующую форму:

выключатели, переключатели, автоматы защиты, предохранители, световые табло и сигнальные лампы. Для работы САПР ПКЭ необходимо наличие файлов с описанием объекта контроля и описанием логики работы элементов. Указанные файлы находятся в базе данных ПОИСК и необходимая информация извлекается специальными средствами СУБД.

Рассмотрим основные правила формирования ПК и алгоритмы работы САПР ПКЭ, чтобы можно было понять принципы автоматизированного проектирования ПК. Весь процесс проектирования программ контроля электросборок условно можно разбить на несколько этапов:

1. Формирование математической модели электросборки по описанию объекта контроля.
2. Формирование групп обмоток реле с одинаковым управлением (фактически определяется параллельное соединение обмоток).
3. Формирование списка обмоток с самоблокировкой.
4. Разбиение объекта контроля на простые цепочки и классификация цепей на цепи управления и монтажные цепи.
5. Анализ цепей управления и формирования для них кадров контроля.
6. Анализ монтажных цепей.
7. Синтез общей программы контроля из ПК отдельных цепей.
8. Формирование измерительных кадров.

Все эти этапы реализованы в САПР ПКЭ. Ниже будут описаны не все, а лишь наиболее важные этапы. Проектирование программ контроля ведется по кадрам. Кадр контроля – элементарная проверка, определяемая совокупностью входных воздействий, подаваемых на объект, и составом контрольных точек, с которых снимаются ответы объекта на это воздействие. Программа контроля (ПК) или тест контроля – это совокупность кадров контроля, обеспечивающих проверку правильности выполненного монтажа в полном объеме. В общем случае ПК – это реализация некоторого алгоритма диагностирования объекта контроля с целью определения состояния последнего. По своему функциональному назначению все кадры контроля разбиты на следующие типы:

- кадры проверки монтажа;
- измерительные кадры;
- кадры контроля сопротивления изоляции.

Кадр проверки монтажа представляет собой набор входных воздействий и набор выходных адресов (совокупность точек, где при заданном наборе входных сигналов должен появиться ответ), а также операции по ручным действиям, если в цепи есть органы ручного управления. Развитие системы контроля и испытаний включает измерительные кадры. Кадр для контроля сопротивления изоляции формируется для цепи типа провод. По своей структуре он совпадает с кадром проверки монтажа, но содержит спец. символ, показывающий, что после контроля монтажа необходимо произвести замер сопротивления изоляции. Измерительный кадр служит для проверки какого-либо параметра цепи: сопротивления, емкости или индуктивности. Дадим некоторые определения, которые нам впоследствии будут необходимы.

Определение 1: Цепь назовем цепью управления, если в ее составе есть обмотка реле.

Определение 2: Цепь управления назовем цепью управления реле с самоблокировкой, если в ней есть контакты того же реле.

При формировании цепей проверяются следующие ограничения, вытекающие из функционирования электросборки:

1. В цепи должно быть не более одной обмотки.
2. Цепь должна содержать элементы однопольные. Например, диоды только в прямом или только в обратном направлении.
3. Для обмотки с самоблокировкой должно быть как минимум 2 разные цепи: одна с самоблокировкой и одна без нее.

Отметим тот факт, что для обмотки реле с самоблокировкой мы должны обязательно построить как минимум одну цепь с самоблокировкой и одну цепь без блокировки, чтобы правильно сформировать алгоритмы проверки,

поэтому еще до начала выделения цепей необходимо определить все обмотки с самоблокировкой. При этом надо учесть, что в сборках могут быть обмотки с одинаковым управлением (параллельное соединение обмоток). Допустим, реле k_1 и k_2 имеют одинаковое управление. Тогда их обмотки можно считать идентичными и условно объединить контактные группы реле k_1 и k_2 . Т.е. если в цепи управления k_1 есть контакты k_2 , то это также будет цепь управления с самоблокировкой. В данном описании не будем останавливаться на алгоритмах определения параллельных обмоток и обмоток с самоблокировкой, чтобы не загромождать описание. Скажем только, что это реализовано на базе нашей ММ. Теперь можно непосредственно перейти к построению цепей.

Согласно определению, цепь – это последовательность ребер, в которой у каждого ребра одна из граничных вершин является граничной вершиной для предыдущего ребра, а другая – граничной вершиной для следующего ребра. Цепь называется простой, если все ее ребра различны. Таким образом, надо найти эффективную процедуру, позволяющую построить простые нециклические цепи на нашем графе $G(P,L)$, причем каждое ребро и каждая вершина должны войти как минимум в одну цепь. Процесс формирования цепей должен продолжаться до тех пор, пока не останется ни одной вершины и ни одного ребра, не вошедшего в цепи. Очевидно, что цепь должна начинаться и оканчиваться в конечной вершине. Представим алгоритм построения цепей:

1. *Выбор входа:* Устанавливаем узел $v(i)$ с максимальным весом. Под весом будем понимать количество единиц в столбце тензора.

2. *Выбор элемента:* Относительно каждой единицы в столбце $v(i)$ находим выходные узлы. Для этого в тензоре строятся все цепи, берущие начало в выбранной точке входа. Каждой единице поставлена в соответствие клемма элемента из множества внутренних монтажных точек (по правилам построения тензора соединений). Запоминаем все эти элементы. Дерево цепей будет выглядеть на данном этапе следующим образом:

$v(i)$ – вход

$e(j(1)), e(j(2)) \dots e(j(k))$

k – число единиц в i -ом столбце; $j(1), \dots, j(k)$ – номера строк, где стоят 1.

$e(m)$ – элемент из m -ой строки тензора.

3. Для каждой точки $e(m)$ находим все внутренние связи (по описанию элемента). При этом, если $e(m)$ – коммутационный элемент, переключающий вход на разные выходы, то фиксируем все соединения. Получим следующее дерево цепей:

$v(i)$ – вход

$e(j(1)) \dots e(j(k))$

$e(l(1, 1))...e(l(1, n)) \dots e(l(2, 1))...e(l(2, n(k)))$

где $l(i, j)$ – номера строк в тензоре для внутреннего соединения элемента $e(i)$; $n(k)$ – число внутренних связей для $e(i)$.

4. Для каждого $e(m)$, полученного на последнем шаге, выполняем следующее: если столбцу, в котором стоит 1, приписано пустое поле, переходим к п.2. В противном случае в цепь заносим выход цепи (то, что приписано столбцу). Цепь считается сформированной. Алгоритм завершается в том случае, когда каждый контакт вошел хотя бы в одну цепь.

В каждом проверяемом объекте цепи, как правило, не разрозненные, а соединенные каким-либо образом. При проверке одной цепи могут появиться сигналы на входах или выходах других цепей. Встает закономерный вопрос: а как же получить общую ПК из ПК отдельных цепей. Очевидно, что простым объединением мы не достигнем нужного нам результата, т.к. полученная таким образом программа может содержать одинаковые кадры, может привести к появлению сигнала на общей шине, а через нее на большом количестве входов, что существенно затрудняет поиск неисправности. Обозначим через $E=\{e(1), e(2), \dots, e(n)\}$ множество всех элементарных проверок, полученное путем объединения тестов контроля цепей. При этом выходная реакция объекта на каждую из элементарных проверок не зависит от того, в какой последовательности элементарные проверки подаются на объект. Множество контролируемых выходов объекта, т.е. тех выходов, значение сигнала на которых определяют реакцию объекта на элементарные проверки, обозначим через $W=\{w(1), w(2), \dots, w(m)\}$.

Основная задача данного этапа – сформировать множество W . При этом допускается

изменение в некоторых проверках совокупности входных сигналов. Это делается для сохранения глубины проверки. Основные правила формирования кадра контроля электросборки покажем на конкретных примерах.

1. Параллельное соединение цепей. При формировании ПК проверяем, можно ли при контроле $k1$ разорвать цепь с $k2$ и наоборот. Если это возможно, то во все проверки для первой цепи добавляем точки, обеспечивающие разрыв второй цепи. Аналогично и для второй цепи. Тем самым обеспечиваем глубину проверки с точностью до одного элемента и требуемую полноту.

2. Появление сигнала на ОШ. При прохождении сигнала "+" на ОШ и через нее на большое число входов электросборки лучшим вариантом исключения такой ситуации является подача минуса на ОШ. Поэтому при синтезе общей ПК в данной ситуации проверяется возможность подачи минуса на ОШ.

3. Корпусные цепи. На этапе формирования кадров цепи нет контроля за ситуацией, приводящей к короткому замыканию (КЗ) и ситуацией, способствующей появлению КЗ в случае неисправности.

4. Объединение проверок. Для данного примера при проверке контакта реле $k1$ получим сигнал на всех входах фрагмента схемы. В таком случае целесообразно объединить проверку $k1$ и лампочки $h1$.

СПИСОК ЛИТЕРАТУРЫ:

1. Гренандер, У. Лекции по теории образов. Синтез образов. Том 1. – М.: Мир, 1979. 383 с.
2. Гилл, А. Введение в теорию конечных автоматов. – М.: Наука, 1966. 272 с.
3. Чжен, Г. Диагностика отказов цифровых вычислительных систем / Г. Чжен, Е. Мэннинг, Г. Метц. – М.: Мир, 1983. 264 с.

THE AUTOMATED SYSTEM OF SYNTHESIS AND ANALYSIS OF CONTROL AND TESTING PROGRAMS OF ONBOARD AUTOMATION COMPLEX DEVICES

© 2014 V.A. Prilepsky, A.N. Koptev, Yu.V. Kovalenko

Samara State Aerospace University named after academician S.P. Korolyov

In article the scientific and technical problem of synthesis the control and test programs of network structures automatic devices of onboard complexes of aircraft electric equipment with application of theory images methods for solution the production problems of estimation the technical state of finished products of electrotechnical production on aircraft constructing enterprises is considered.

Keywords: *synthesis, analysis, system, program, control, tests, device, onboard electrotechnical equipment*

Vasily Prilepskiy, Candidate of Technical Sciences, Associate Professor at the Aircraft Maintenance Department. E-mail: 2pr@inbox.ru; Anatoliy Koptev, Doctor of Technical Sciences, Professor, Head of the Aircraft Maintenance Department. E-mail: eat@ssau.ru; Yuliya Kovalenko, Post-graduate Student