

АВТОМАТИЧЕСКИЙ СИНТЕЗ ФУНКЦИОНАЛЬНЫХ СХЕМ

Наталья Филатова, Олег Ахремчик, Олег Куприянов

Аннотация: Доклад посвящается интеллектуализации компьютерной обработки информации в области автоматизированного проектирования систем управления. Основой автоматического синтеза служат модели структуры и описания составляющих ее основных элементов. Алгоритмы автоматического синтеза схем являются результатом применения технологий инженерии знаний и поиска данных при определении отношений порядка. Связи на структурной схеме детализируются посредством проверки условий совместимости и установления дополнительных элементов для их истинности. Представляется клиент-серверная архитектура программного обеспечения для автоматического синтеза функциональных схем. Основным ее компонентом является база данных. Задания, схемы и описания элементов составляют ее декларативную часть. Правила синтеза – процедурную часть. Количество генерируемых вариантов ограничивается. Апробация программного обеспечения доказывает возможность автоматического синтеза целей систем управления в ходе разработки функциональных схем.

Ключевые слова: представление знаний, САПР, автоматический синтез.

1. Задача автоматического синтеза функциональных схем

На современном этапе развития САПР систем управления остается не решенной проблема автоматического синтеза схем. Существующие программные средства автоматизированного проектирования поддерживают решение этих задач в интерактивном режиме. Все процедуры выбора, оценки варианта технического решения выполняет инженер. В этой ситуации существенно ограничивается число анализируемых альтернативных вариантов схем, что приводит к снижению качества проектного решения.

Все технологии разработки схем систем управления включают четыре взаимосвязанных этапа (табл.1). Каждый этап завершается созданием определенного типа схемы. Выделение отдельных этапов в решении этих задач связано с блочно-иерархическим представлением объекта проектирования (ОП) в САПР [1, 2].

Таблица 1 позволяет обосновать требования к структуре знаний предметной области. Действительно, при создании структур, моделирующих схему рассуждения эксперта, разрабатывающего схему системы автоматизации (СА), можно ограничиться знаниями, охватывающими первые 4 вертикальных уровня проектирования. Знания, связанные с методами создания блоков, (охватывают вторые 4 вертикальных уровня проектирования) не будут востребованы при разработке систем СА, как и аналогичные структуры третьего уровня.

Таблица 1.

Объект проектирования	Уровни проектирования (типы схем)			
	1	2	3	4
Система автоматизации (СА)	Структурная схема (СС)	функциональная схема (ФС)	Принципиальная электрическая схема (ПЭС)	Монтажно-коммутационная схема (МКС)
Блок / устройство	- " -	- " -	- " -	- " -
Элемент блока	- " -		- " -	- " -

Каждому уровню проектирования соответствует, по меньшей мере, одна модель объекта проектирования (схемы СА). В САПР схем СА должна поддерживаться иерархия схемных моделей вида : $m_{CC} \Rightarrow m_{FC}$; $m_{FC} \Rightarrow m_{PES}$; $m_{PES} \Rightarrow m_{MKS}$. Процедурные знания САПР должны охватывать методы преобразования всех выделенных моделей объекта проектирования.

Вертикальные уровни декомпозиции объекта проектирования позволяют выделить семантические группы предметных отношений. Каждая такая группа включает отношения, связанные с выполнением одной или нескольких операций. Горизонтальные уровни декомпозиции объекта проектирования позволяют выделить его структурные особенности, которые должны найти отражение в концептуальной структуре предметной области. На основе таблицы можно сделать вывод, что объект проектирования "схема СА" приводит к иерархии трех классов сущностей:

- система автоматизации, заданная на множестве промышленных систем;
- технический блок (устройство), определенный на множестве общепромышленных технических средств автоматики (ТСА);
- элемент блока, определенный на множестве типовых элементов замены.

Как показывает опыт создания интеллектуальных САПР, любая модель j -ого уровня проектирования опирается только на соответствующую модель $(j+1)$ -ого уровня [3]. Поддерживать связи моделей j -ого уровня с моделями $(j+2)$ -ого уровня не целесообразно. Следовательно, в концептуальной структуре знаний рассматриваемой САПР можно ограничиться двумя первыми классами сущностей - схема СА и блок.

Полученные выводы были учтены при разработке системы Functional Design of a Control System (FDcS) для автоматизированной поддержки начальных этапов проектирования систем автоматизации технологических объектов. Главное назначение этой системы – генерация множества вариантов технических решений (в виде функциональных схем автоматизации) при слабых ограничениях на область поиска вариантов.

Постановка задачи синтеза схемы СА и методы ее решения зависят от уровня проектирования. Разработка структурной схемы выполняется с использованием методов теории автоматического управления, и поддерживается множеством известных программ моделирования систем. Построенную модель m_{CC} можно рассматривать как единственное решение, являющееся техническим заданием для всех последующих этапов.

При формировании функциональной схемы решается вопрос технической реализации системы автоматизации, в схему включают блоки технических средств автоматики (ТСА), выпускаемые промышленностью. На основе одной структурной схемы можно создать несколько различных вариантов функциональных схем (ФС), отличающихся по стоимости и некоторым эксплуатационным

характеристикам. Учитывая большую размерность множества и постоянные изменения на рынке ТСА, инженер проектировщик обычно ограничивается одним вариантом функциональной схемы.

Переход от структурной схемы к функциональной означает конкретизацию теоретического решения на базе некоторого комплекса технических средств. Если ТСА не единственно, а представлено конечным множеством блоков, то теоретически можно получить все возможные варианты функциональных схем:

$$m_{CC} \Rightarrow m_{FC1}, m_{FC2}, \dots m_{FCN},$$

причем количество схем (N) будет зависеть только от объема БД ТСА.

Модели m_{CC} и m_{FC} различаются между собой степенью детализации описания объекта (схемы автоматизации). При переходе от модели m_{CC} к модели m_{FC} увеличивается как число блоков и связей, так и число характеристик, используемых в описаниях блоков.

В рамках каждой модели схема рассматривается как совокупность (n) контуров. Каждый контур состоит из цепей, которые могут в общем случае иметь разную длину.

$$Sh = \bigcup_{i=1}^n Kon_i, \quad (\forall i) Kon_i = \bigcup_{j=1}^{m_i} Ze_j, \quad \bigcap_{j=1}^{m_i} Ze_j \neq \emptyset \quad (1)$$

m_i - число цепей в i-ом контуре

Любая цепь, в свою очередь, состоит из одной или более связей.

$$Ze_j = S_{1,2}^j \circ S_{2,3}^j \circ \dots \circ S_{g,K_j}^j, \quad S_{r,t}^j = S^j(B_r, B_t) \quad B_r, B_t \in B \quad (2)$$

где: k_j – число связей в j-ой цепи (длина цепи), $S(B_r, B_t)$ – связь блока g-ого с блоком t-ым, B – множество блоков ТСА.

Порядок следования связей между элементами цепи строго определен, поэтому цепь представляет последовательность связей, а не совокупность. Каждую связь будем рассматривать как пару элементов (блоков), которые находятся в отношении “соединять”:

$$R^S = \langle \text{соединять}, x_i, y_r \rangle \quad (3)$$

где x_i – выход блока B_i , y_r – вход блока B_r .

Тогда отношение $(x_i R^S y_r)$ соответствует высказыванию “выход i-ого блока соединен с входом r-ого блока” и формируется связь $S(B_i, B_r)$. Каждую связь можно рассматривать как элементарный орграф, имеющий вершину-исток (Is) и вершину-сток (St). Для связи $S(B_r, B_t)$ соответственно $Is = B_r$, $St = B_t$.

Для построения отношения порядка (Ze) на множестве связей необходимо, чтобы для каждой пары связей выполнялось требование:

$$S_i(Is, St) \cap S_j(Is, St) \rightarrow St_i = Is_j \quad (4)$$

В множестве блоков ТСА (B) можно выделить три подмножества: измерительных преобразователей ($ИП=B^1$), исполнительных устройств ($ИУ=B^3$) и подмножество B^2

$$B^2 = B \setminus (B^1 \cup B^3) \quad (5)$$

Блоки из B^1 , B^3 выбирают исходя из требований технологов к вектору входов (выходов) объекта управления. Другие блоки (из B^2) выбирают с учетом характеристик B^1 , B^3 .

Операции выбора блоков ТСА для заданной структурной схемы относят к этапу разработки ФС, однако, при создании САПР <<FDcS>> было введено допущение:

- считать, что для каждого элемента цепи структурной схемы, выполнена операция назначения, определив название блока ТСА.

Тогда при создании ФС задача выбора блоков для построения Ze будет решаться на множестве B^2 , что приведет как к уменьшению размерности задачи перебора, так и к точному заданию истоков и стоков всех цепей ФС.

При введенных допущениях модель ФС будет отображением m_{CC} на множестве концептов ТСА, включенных в цепи ФС (рис.1).

Рассмотрим на примере взаимосвязь между моделями схемы. На 1-ом уровне проектирования структурную схему можно представить в виде графа ($G:: IP \rightarrow P \rightarrow IU$). Гомоморфное отображение этой структуры на таксономию "ТСА" позволяет получить модель m_{CC} и ее граф $G1:: (IP=ТСП) \rightarrow (P=БК1/Р130) \rightarrow (IU=МЭО)$. Между моделями G и $G1$ существует гомоморфизм. При разработке ФС на основе графа $G1$ формируется граф $G2$.

Очевидно, отображение вида $G2$ (или m_{FC}) построено правильно, если объединение $G1$ и $G2$ даст транзитивное замыкание $G2$ на вершинах из $G1$.

Рассмотрим формальную постановку задачи построения множества моделей функциональных схем на основе единственного варианта структурной схемы.

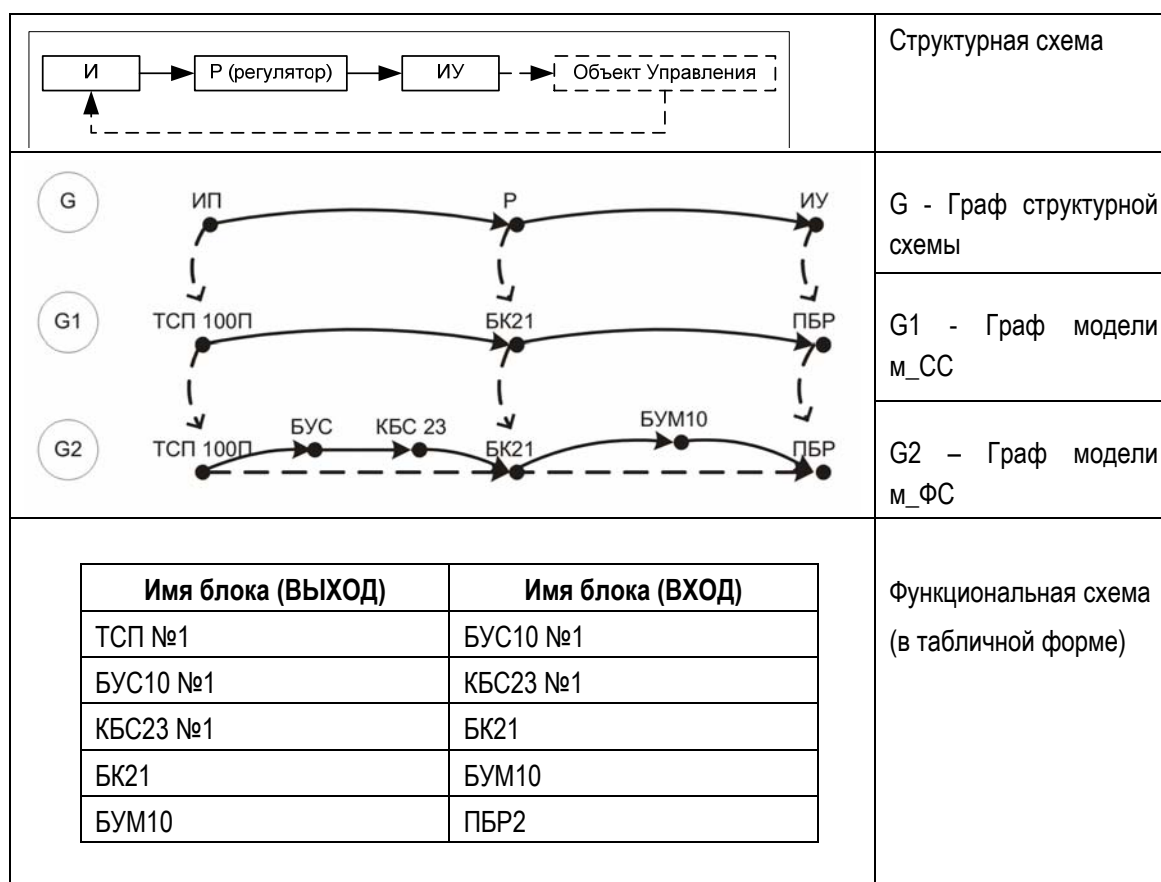


Рис.1. Связи между структурной и функциональной схемами и их моделями.

Пусть имеется модель m_{CC} , заданная n контурами, каждый контур состоит из m_i цепей (1). Введено ограничение на длину каждой цепи ($k_j=1$). Тогда вместо (2) каждая цепь вырождается в связь.

$$(\forall j) Ze_j = S^j(x_r, y_t), \quad B_r, B_t \in B \quad (6)$$

где: x_r – выходы блока B_r , y_t – входы блока B_t .

Каждый блок (B_i) может быть представлен множеством атрибутов, которое можно представить в виде нескольких подмножеств:

$$B_i :: A_i = (A_{i1}, A_{i2}, \dots, A_{ip}) = \bigcup_{p=1}^{pp} A_{ip} \quad (7)$$

В зависимости от размерности (pp) множества A_{ip} можно создавать различные по степени детализации описания технических блоков. Для иерархии схем автоматизации будет справедливо:

$$B_i^1 \subset B_i^2 \subset B_i^3, \text{ но тогда } A_i = \bigcup_{p=1}^3 A_{ip} \quad (\forall p) A_{ip} \subseteq A_i, \bigcap_{p=1}^3 A_{ip} = 0 \quad (8)$$

где: B_i^1, B_i^2, B_i^3 – (соответственно) описания i -ого блока для структурной, функциональной и принципиальной электрической схем.

При создании модели m_CC используют описания блоков (B_i^1) с минимальным числом атрибутов. В рассматриваемой постановке задачи множество A_{i1} , включает названия блока, модели и выпускающей фирмы.

Необходимо на основе модели m_CC вида (1,6,7) найти такое множество функциональных схем $\{m_FC_i\}$, в котором для каждой m_FC_i выполняются условия:

- а) число и состав контуров не меняется при переходе от m_CC к любой m_FC_i ,
- б) число цепей не меняется при переходе от m_CC к любой m_FC_i ,
- в) для каждой связи (S_i), входящей в цепь из модели m_FC , должны выполняться ограничения на вид функциональных преобразований и уровень сигнала, которые обеспечивает указанная электрическая связь:

$$(Ogr_fun(B_r) = Ogr_fun(B_l)) \& (Ogr_DSign(B_r) \leq Ogr_DSign(B_l)) = A, \quad (9)$$

$$A \Rightarrow S(B_r, B_l) \in m_FC$$

- г) каждая цепь функциональной схемы имеет одинаковые с соответствующей цепью в структурной схеме истоки и стоки:

$$Ze_i \in m_CC, Ze_i = S_i(x_r, y_l), Ze_i^* \in m_FC, Ze_i^* = S_1 \circ \dots \circ S_k, Ze, \quad (10)$$

$$S_1 \cap S_i = x_r, S_k \cap S_i = y_l$$

Переход от m_CC к m_FC осуществляется путем последовательной проверки ограничений (9) для всех цепей CC . Возможны три варианта результатов проверки:

- 1) Если при анализе Ze вида (6) условие (9) выполняется, то $S^j(x_r, y_l)$ переносится из m_CC в m_FC без изменений.
- 2) Если для некоторой цепи Ze из m_CC условие (9) не выполняется. Тогда на множестве блоков B^2 необходимо найти такой B_l , что при его подстановке в (9) вместо B_r условия (9) будут выполняться. Тогда формируется новая связь $S^j(B_r, B_l)$. Модель m_FC принимает вид:

$$Ze_j = S^j(B_r, B_l) \circ SG(B_l, B_l) \quad (11)$$

Таким образом связь $S^j(x_r, y_l)$ из m_CC заменяется двумя новыми: скорректированной связью $S^j(B_r, B_l)$ и связью-гипотезой $SG(B_l, B_l)$.

Расщепление связи происходит по правилу: исток в скорректированной связи равен истоку в родительской связи, сток связи-гипотезы равен стоку в родительской связи:

$$Is_{rt} = Is^{kor} \quad St_{rt} = St^{gip} \quad (12)$$

Скорректированная связь включается в м_ФС, для гипотетической связи повторяются операции проверки условий (9) и при необходимости коррекции путем расщепления.

3) Если хотя бы для одной цепи после завершения цикла коррекции не удастся выполнить условия (9-10), то в необходимо вносить изменения в модель м_СС.

2. Система проектирования функциональных схем "FDcS"

Программное комплекс (ПК) системы проектирования функциональных схем сочетает в себе функции клиентской части системы управления реляционной базой данных, графического редактора структурных схем и клиентскую и серверную части реализации алгоритма синтеза функциональных схем.

Разработка программы выполнена на языке C++ с использованием методов объектно-ориентированного программирования, библиотеки стандартных шаблонов STL, мультиплатформенной C++ библиотеки для разработки графического интерфейса приложений (GUI) QT от Trolltech, динамически подключаемой библиотеки libpq.dll, поставляющей интерфейс с сервером базы данных PostgreSQL.

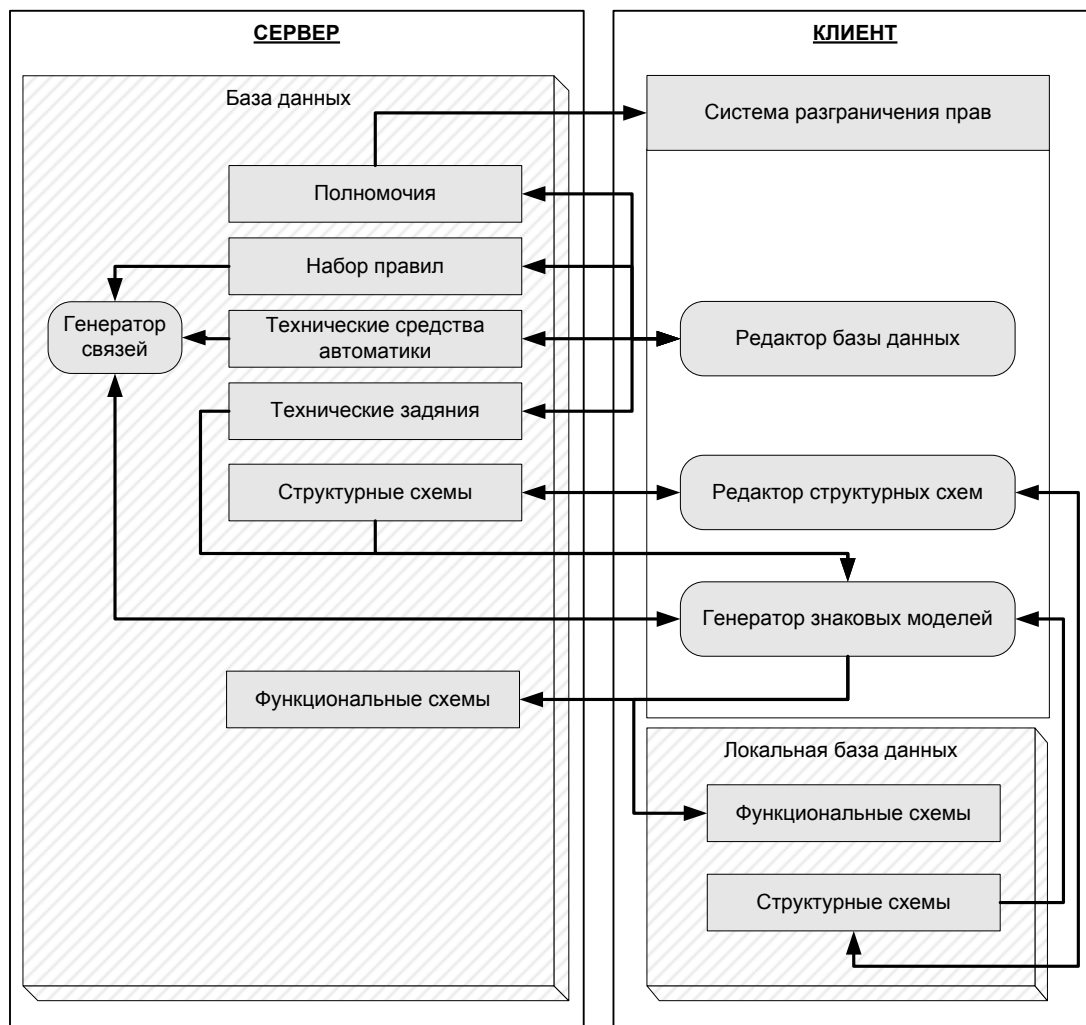


Рис.2. Архитектура системы проектирования функциональных схем "FDcS".

ПК имеет модульную, распределенную архитектуру, которая позволяет успешно функционировать в сетях различного масштаба, как в корпоративных (Intranet), так и в сети Internet. Допустима установка

программного обеспечения на одной рабочей станции, в этом случае на нее возлагаются дополнительные функции сервера баз данных.

Система распределения прав между группами пользователей позволяет защитить информацию в базе данных от действий неквалифицированных пользователей, а также ограничить доступ к административным функциям ПК. Архитектура системы представлена на рис.2.

Главным серверным компонентом ПК является база данных (БД). База данных включает декларативные и процедурные знания предметной области. Декларативную часть БД образуют данные по техническим средствам автоматизации, техническим заданиям, структурным схемам и сгенерированным функциональным схемам. К процедурной части БД относятся набор правил фокусировки поиска подстановок в продукции вида (9) и процедуры, образующие серверную часть модуля генерации знаковых моделей – генератор связей.

Клиентская часть ПК включает модули редактора БД, графического редактора структурных схем, генератора знаковых моделей и локальной БД структурных и функциональных схем. Все они работают под контролем модуля системы разграничения прав.

Модуль генератора знаковых моделей в качестве исходных данных получает макет технического задания с вербальным описанием требований к системе управления и ее структурной схемой. Синтез очередного варианта функциональной схемы осуществляется путем совместного взаимодействия модулей генератора знаковых моделей и генератора связей.

Ядром информационного обеспечения системы "FDcS" является база данных технических средств автоматизации - БД ТСА. Все множество ТСА по функциональному признаку разбиты на группы, составляющие иерархическую структуру (дерево) без множественного наследования [4]. В используемой реляционной базе данных (PostgreSQL) для хранения такого рода иерархии необходимо добавить в описание группы элементов дополнительное поле – ссылку на первичный ключ родителя.

Для каждого элемента указывается уникальный идентификатор, наименование, идентификатор группы, к которой принадлежит элемент (поле "tree_node") и массив, хранящий полный путь (поле "path") по иерархии групп начиная от корневой группы и заканчивая группой, к которой принадлежит элемент.

Дополнительное описание пути является избыточным, т.к. его всегда можно определить рекурсивно. Однако для быстрого выбора элементов, относящихся к определенной группе (не только указанной, но и всех ее наследников), рекурсивный способ не подходит из-за недостаточного быстродействия.

Скорость генерации цепей функциональной схемы определяется затратами времени на поиск допустимых подстановок в правило (9), выполняемых генератором связей. Для уменьшения времени генерации цепей при большом количестве элементов таксономии ТСА используются специальные правила фокусировки, сужающие область поиска элементов.

Введено два типа правил: ИСКАТЬ и НЕ ИСКАТЬ, которые задаются для пары групп элементов (G_1, G_2). Например, в соответствии с правилом <ИСКАТЬ(G_1, G_2)> при построении связи $S(B_i, B_j)$ если $B_i \in G_1$ или любой ее дочерней группы, поиск B_j для подстановки в (9) ограничивается на таксономии ТСА группой G_2 или любой ее дочерней группой. Правило <НЕ ИСКАТЬ(...), G_2 > позволяет вести поиск допустимых подстановок в (9) среди всего множества элементов, за исключением элементов указанной группы (G_2).

Приведем пример использования правил. Для указанных узлов иерархии групп элементов

- измерительные преобразователи (ИП)
- └─ нормирующие преобразователи (НП)

└─ первичные преобразователи (ПП)

можно ввести правило RULE 2 :: <ИСКАТЬ(ПП, НП)>. Интерпретация RULE 2 :: Поиск дополнительных устройств для подключения к первичным преобразователям (датчикам) необходимо вести только в группе нормирующих преобразователей, преобразующих сигналы датчиков в сигналы унифицированных диапазонов.

Другой важной проблемой генерации функциональных схем является резкое увеличение числа возможных вариантов схем при увеличении размерности множества блоков ТСА. Для решения этой проблемы в системе "FDcS" используются две стратегии: ограничения общего количества требуемых функциональных схем и исключение из процесса генерации технически не реализуемых схем.

При большом количестве заданных в БД блоков ТСА возможен случай, когда в процессе генерации цепи происходит многократная коррекция связи путем расщепления, для чего может использоваться один и тот же блок ТСА. Это может привести к зацикливанию и образованию цепочек связей вида В1–В2–В3–В1–В2–В3–В1... В тоже время известно, что на практике цепи, состоящие из пяти и более связей, обычно не используются из-за низкой надежности, высокой сложности и стоимости. Для исключения зацикливания введено ограничение на количество связей в цепи (m – настраиваемый параметр). При обнаружении в цепи более (m) связей процесс генерации цепи прекращается, а сама схема помечается как технически не реализуемая.

Заключение

Создание системы Functional Design of a Control System (FDcS) доказывает возможность перехода от параметрического синтеза структурных схем, отражающих динамику АСУТП, к автоматическому структурному синтезу функциональных схем в ряде измерительных и управляющих цепей. Модели, положенные в основу построения баз данных и знаний являются базой для промышленной технологии начальных стадий автоматизированного проектирования АСУТП.

Библиография

- 1.Норенков И.П., Введение в автоматизированное проектирование технических устройств. Москва, 2005.
- 2.Цветков В.Д. Системно-структурное моделирование и автоматизированное проектирование технологических процессов. Минск. 1990.
- 3.Филатова Н.Н. Автоматическое формирование знаний в САПР. Тверь. 1996.
- 4.О.Л.Ахремчик. Программная система функционального проектирования аппаратных комплексов АСУТП. / Программные продукты и системы, N2. 2007.

Информация об авторах

Филатова Наталья Николаевна - профессор, Тверской государственный технический университет, 170026, Ст. Никитина 22, Тверь, Россия; e-mail: nfilatova99@mail.ru

Ахремчик Олег Леонидович - доцент, Тверской государственный технический университет, 170026, Ст. Никитина 22, Тверь, Россия; e-mail: ivanova@foibq.com

Куприянов Олег Владимирович - магистр, Тверской государственный технический университет, 170026, Ст. Никитина 22, Тверь, Россия; e-mail: aven@tvcom.ru