

УДК 628.440.22

**А.В. Григорьев**

Донецкий национальный технический университет, г. Донецк  
кафедра прикладной математики и информатики  
E-mail: grigorie@r5.dgtu.donetsk.ua

## **МЕТОДЫ РЕШЕНИЯ ЗАДАЧИ СТРУКТУРНОГО СИНТЕЗА В ИНТЕЛЛЕКТУАЛЬНЫХ САПР, ПОСТРОЕННЫХ НА ОСНОВЕ СЕМИОТИЧЕСКОЙ МОДЕЛИ СТРУКТУР**

### **Аннотация**

*Григорьев А.В. Методы решения задачи структурного синтеза в интеллектуальных САПР, построенных на основе семиотической модели структур. В работе описывается формальная постановка задачи синтеза структур на основе семиотической модели структур, а так же методы решения данной задачи. Набор методов решения задачи обеспечивает адаптацию на уровень квалификации эксперта в данной предметной области. Каждый режим предполагает соответствующий способ построения базы знаний и свой метод организации вывода.*

**Ключевые слова:** семиотическая модель структур, формальная постановка задачи, модуль знаний, И-ИЛИ-дерево, продукции, адаптация на квалификацию эксперта, режимы вывода..

### **1. Общая постановка проблемы.**

Проблема предметной адаптации в САПР в настоящее время является одной из самых актуальных и важных. В настоящее время наблюдается процесс резкого расширения числа предметных областей, где требуется создание конкретного проблемно-ориентированного САПР, с одновременным сокращением периода жизни самих предметных областей, усложнением и сокращением сроков процесса проектирования. При этом требования к качеству проектирования и эффективности готовых решений САПР только возрастают. На рис. 1. представлен график динамики жизненных циклов технологий создания технических изделий, построенный в соответствии с данными, подготовленными сетью по продаже электроники Richmania [1]. Все технологии упорядочены по дате появления. Спецификой данного процесса является прогрессирующее сокращение периода жизни технологий при одновременном увеличении числа вновь появившихся технологий, что особенно резко проявляется в настоящее время. Каждой из перечисленных технологий соответствует некоторый тип изделия и, в общем случае, соответствующие средства проектирования, т.е. САПР. В области проектирования характер процесса динамики жизненных циклов предметных областей объективно выражается как в появлении новых САПР, так и в сокращения периода появления новых версий существующих САПР, при одновременном их усложнении, расширении комплекса средств их предметной адаптации. Такая ситуация делает актуальной задачу построения более эффективных инструментальных средств (платформ) как для автоматизированного построения новых высокоэффективных проблемно-ориентированных САПР, соответствующих той или иной новой технологии, так и – для повышения уровня средств предметной адаптации для существующих САПР. Цель такого решения задачи – это сокращение трудоемкости и времени как создания новых САПР так и адаптации существующих САП, при одновременном обеспечении повышения их эффективности.

**2. Постановка задачи исследования.**

Задача построения инструментальных средств для создания интеллектуальных систем структурного синтеза так же являются важной и актуальной. Способы решения задачи синтеза структур в интеллектуальных САПР могут существенно зависеть от: специфики структур в данной предметной области; уровня квалификации эксперта по знаниям, выполняющим построение базы знаний САПР в данной предметной области.

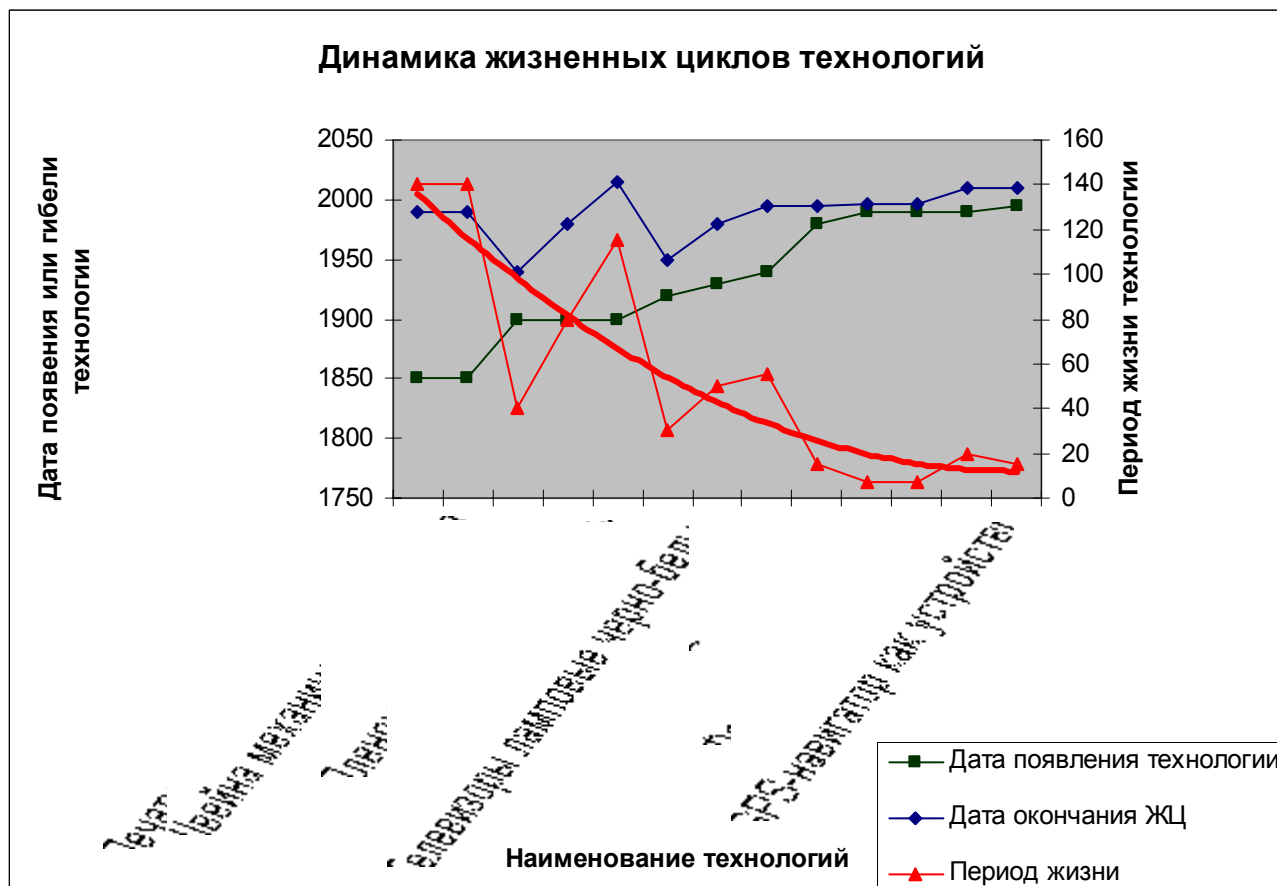


Рисунок 1 – Специфика динамики жизненных циклов технологий в современный период

Ранее автором была предложена унифицированная модель структур, адаптируемая с как специфике любой предметной области, так и – к уровню абстрактного представления структур [2,3]. На основе данной модели структур была предложена интеллектуальная модель САПР, приведенная в [4]. Суть подхода [4] состоит в формировании на основе множества текстовых описаний моделей прототипов в форме И-ИЛИ-дерева и формирования ряда продукций над ИЛИ-узлами (синтермами), связывающими имеющий смысл в данном наборе прототипов комбинации признаков. В упомянутой работе для описания зависимостей вариантов была применена модальная логика («необходимо», «возможно»). Данный подход делает возможным синтез необходимой структуры по ряду требуемых структурных отличий, т.е. путем выбора требуемых альтернатив в тех или иных ИЛИ-узлах И-ИЛИ-дерева.

Предложенная модель может быть рассмотрена с точки зрения теории сложности систем автоматического управления [5], как модель САПР для решения типичных задач проектирования, где выбор решения из целевого пространства систем (ЦПС) осуществляется путем определения необходимого технического задания из пространства обликов систем (ПОС), рассматриваемого как множество альтернатив из ИЛИ-узлов И-ИЛИ-дерева. В случае неоднозначности полученного решения применяется критерий сложности, обеспечивающий выбор решения, оптимального в смысле некоторой R-эквивалентности, что

в данном случае означает определение наиболее приемлемой альтернативы в некотором ИЛИ-узле, оставшимся недоопределенным.

Предложенная модель рассматривается как семиотическая, т.е. открытая формальная модель структур (СМ) [6]. СМ представляет собой ряд формальных моделей, связанных отношениями достижимости, представляющими собой изменения элементов формальных моделей. На основе данной модели САПР была предложена СМ множества моделей структур, связанных отношениями зависимости [7]. В рамках данной СМ множества структур были построены:

- сигнатура  $T$ , для чего определен ряд новых категорий, входящих в сигнатуру;
- множество синтаксических верных выражений, построенных на базе элементов сигнатуры, т.е. грамматика  $G$ ;
- аксиоматика  $A$ , как часть множества синтаксических верных выражений;
- множество правил вывода  $P$ , позволяющих проверить, относится ли данное синтаксически верное выражение к множеству семантически верных выражений  $S$  (множеству прототипов).

Данная СМ имеет такие правила изменения:

- сигнатура  $T$ , множество синтаксических верных выражений  $G$ , аксиоматика  $A$ , множество правил вывода  $P$  – предполагаются неизменными;
- множество семантически верных выражений  $S$  (множество прототипов) – открыто, т.е. предполагает ввод новых прототипов.

Постановка и метод решения задачи построения правил изменения множества семантически верных выражений  $S$  (множество прототипов), представленное И-ИЛИ-деревьями, описано в [8]. Принципиальным отличием предложенного подхода построения интенциональной составляющей модели, т.е. множества интенциональных отношений, является построение идентификации на основе идентификации комплекса внешней и внутренней среды именуемого объектом. Подобный подход может быть определен как «интегральная идентификация» объектов. Соответственно, используя данный подход, можно определить:

- интегральное свойство (как параметр);
- интегральный идентификатор блока и т.д.

Таким образом, был определен единый механизм идентификации структур, функций и состояний объектов в условиях замкнутых ограниченных моделей. Для применения предложенной СМ структур в практике синтеза структур необходимо дополнительно решить следующие задачи:

- 1) Выполнить развитие приведенных ранее положений, включая:
  - разделение единого модуля описания модели объекта на различные подмодули (отдельные И-ИЛИ-деревья) описания: внешней границы, внутренне границы, внутренней среды (функция как частный случай внутренней среды), отдельных групп связей, групп свойств и т.д.;
  - введение сложной идентификации элементов модели, а не только прототипов в целом.
- 2) Предложить формальную постановку задачи синтеза структур на основе данной расширенной семиотической модели;
- 3) Предложить методы решения поставленной задачи синтеза, адаптируемые к уровню квалификации эксперта в данной предметной области.

### **3. Решение задач и результаты исследований.**

Выполним решение поставленных выше задач.

#### **3.1. Формальная постановка задачи синтеза структур**

Предложенная СМ должна быть рассмотрена с точки зрения особенностей ее как базы знаний, разделенной на модули знаний в соответствии с числом типов структурных блоков.

Структура блока некоторого  $t-1$ -го уровня иерархии (номер уровня определяется по внутренней границе блока) идентифицируется пятеркой:

$I_{t-1} = (W_t(M_{t-1}, n_{t-1}, G_{t-1}); V_t(R_t, S_t))$ , где:

1)  $W_t$  - комплекс идентификаторов внешней среды блока, включая:

- $M_{t-1}$  - назначение (массив) блоков во внешней среде (вышележащем блоке);
- $N_{t-1}$  - номер блока в массиве;
- $G_{t-1}$  - идентификатор варианта внешней границы блока (как набор данных);

2)  $V_t$  - комплекс идентификаторов внутренней среды блока, включая:

- $R_t$  - идентификатор варианта внутренней границы;
- $S_t$  - идентификатор варианта внутренней среды (как набор связей).

Внутри блока имеет место зависимость:  $S_t \rightarrow \{I_t^k\}$ ,  $k=1, n$ .

Тут  $I_t^k$  - это идентификация  $k$ -го подблока из внутренней среды  $S_t$ , но определенная частично, т.е. - только на уровне идентификаторов внешней среды для данного подблока.

Собственно зависимость реализуется как процесс косвенного синтеза внешней идентификации для всех внутренних подблоков, входящих во внутреннюю среду. Данный синтез выполняется путем объединения воедино, т.е. в  $G_t$ , всех внешних данных, упомянутых в структурных связях внутренней среды  $S_t$  для  $k$ -го подблока с заданным комплексом идентификаторов внешней среды, т.е. с:  $W_t^k = (M_t^k, n_t^k, *)$ .

На основе анализа количества массивов данных, размера того или иного массива, а так же роли массива - вход или выход (чужие или собственные потенциалы), формируется идентификатор варианта внешней границы  $k$ -го подблока  $G_t^k$ :

$G_t^k = \{M_t^{ki}, R_t^{ki}, N_t^{ki}\}$ ,  $i=1, P^k$ , где:

- $M_t^{ki}$  - идентификатор массива данных;
- $N_t^{ki}$  - размер  $M_t^{ki}$  - го массива данных;
- $R_t^{ki}$  - роль массива данных (вход или выход).

В результате синтеза идентификация некоторого подблока из внутренней среды  $S_t$  приобретает вид:

$I_{t+1}^k = (W_{t+1}^k(M_{t+1}^k, n_{t+1}^k, G_{t+1}^k); V_{t+1}^k(*, *))$ , где:

\* - неопределенные значения.

Т.е. при определенной внешней идентификации подблока пока остается неопределенной внутренняя идентификация подблока. При этом  $I_{t+1}^k$  рассматривается как идентификатор модуля знаний о внутренней структуре блока.

**Тип модуля знаний** идентифицируется по составу «собственных», системообразующих свойств во внешней границе блока, т.е. - в  $G_t$ .

Т.о., **основные параметры для работы данного модуля** знаний задаются тройкой:

$W^k = (M^k, n^k, G^k)$ .

Соответственно, при этом могут изменяться:

- идентификатор массива, куда включен данный блок -  $M^k$ ;
- номер блока в массиве -  $n^k$ ;
- состав свойств во внешней границе -  $G^k$ .

Содержимое модуля знаний определяется алгоритмом, который по конкретному варианту внешней идентификации подблока осуществляет выбор внутренней идентификации подблока, т.е.:  $V_{t+1}^k(*, *)$ .

Т.о., фактически рассматривается постановка задачи синтеза внутренней структуры подблока  $V_{t+1}^k$ , заданного модулем знаний  $I_{t+1}^k$  некоторого типа, определяемого в основном составом  $G_t$ , по данной внешней идентификации блока  $W_{t+1}^k$ .

Синтез внутренней идентификации подблока  $V_{t+1}^k$  выполняется в два этапа:

- сначала выполняется синтез внутренней границы подблока  $R^k_{t+1}$ , включая список его внутренних подблоков;

- затем выполняется синтез внутренней среды подблока  $S^k_{t+1}$  как набора связей с участием подблоков, входящих во внутреннюю границу.

В процессе синтеза на каждом этапе вводится ряд **дополнительных параметров**, идентифицирующих особенности внутренней структуры желаемого блока.

Т.о., общая формальная постановка задачи синтеза структуры по модулю знаний включает: тип модуля; основные параметры; дополнительные параметры. В случае неоднозначности полученного решения применяется критерий сложности, обеспечивающий выбор решения, оптимального в смысле некоторой R-эквивалентности, т.е. - определение наиболее приемлемой альтернативы в некотором ИЛИ-узле, оставшимся недоопределенным.

### 3.2. Условие окончания синтеза, базовый уровень

Для базового блока выполняется условие:

$I^k = (W=(M,N); V=(G,G,*))$ , т.е.:

- 1) внешняя и внутренняя границы блока совпадают;
- 2) внутренняя среда блока отсутствует.

### 3.3. Идентификация структур при синтезе

При переходе (синтезе) во внутреннюю среду подблоков выполняется переход:

$V^k=(Gk,*,*) \rightarrow V^{k'}=(G^k, R^k, S^k)$ ,

или

$I_k(W^k(M^k, N^k); V_k(Gk, *, *)) \rightarrow I_{k'}(W^k(M^k, N^k); V^{k'}(G^k, R^k, S^k))$ .

Соответственно, доопределяется идентификатор внутренней среды вышележащего блока:  $S' \leftarrow \{I^{k'}\}, k=1, n$ .

И далее, для вышележащего блока:  $I' \leftarrow S'$ .

Тут  $I'$  – новая, более уточненная идентификация вышележащего блока.

Т.о., динамически формируется следующая цепочка идентификаторов блока глубиной декомпозиции  $L: \{I_l\}, l=1, L$

### 3.4. Методы решения задачи синтеза

Порядок выполнения указанных этапов синтеза, а так же состав параметров для синтеза внутренней структуры на данных этапах различны для различных режимов построения модуля знаний.

Фактически данные режимы построения модулей знаний – это есть различные способы решения задачи синтеза, описанной выше в общем виде.

Будем различать различные режимов построения модуля знаний в зависимости от уровня квалификации эксперта [9]. Уровни квалификации эксперта в данной предметной области при избранном подходе к созданию базы знаний отличаются его возможностью построить ЦПС (множество возможных решений-прототипов) и ПОС (множество ТЗ).

Возможны варианты, когда пользователь может построить: 1) ЦПС и ПОС; 2) только ЦПС; 3) только ПОС; 4) ничего.

Рассмотрим требования, предъявляемые к пользователю, и требования, предъявляемые к инструментальной системе построения интеллектуальных САПР, в каждом конкретном случае:

- 1) Пользователь способен построить ЦПС и ПОС.

Предполагает варианты построения:

- пользователь вводит только набор слов, составляющих ПОС и/или ЦПС; данный случай предполагает автоматическое создание грамматик для ЦПС и/или ПОС на базе набора слов; в этом случае используются неявные продукции;

- пользователь сам вводит грамматики для ПОС и ЦПС; продукции вводятся пользователем явно; это конечный вариант.

- 2) Пользователь способен построить только ЦПС.

Предполагает такие варианты построения ЦПС:

- пользователь вводит набор слов, составляющих ЦПС, что предусматривает дальнейшее автоматическое создание грамматики ЦПС на базе набора слов; используются неявные продукции;

- пользователь вводит грамматику для ЦПС; продукции вводятся пользователем явно; это конечный вариант.

Предполагаемый вариант построения ПОС:

- отсутствующий ПОС строится автоматически на базе грамматики, задающей ЦПС.

3) Пользователь способен построить только ПОС.

Предполагает такие варианты построения ПОС:

- пользователь вводит набор слов, составляющих ПОС, что предусматривает дальнейшее автоматическое создание грамматики ПОС на базе набора слов; используются неявные продукции;

- пользователь сам вводит грамматику для ПОС; продукции вводятся пользователем явно; это конечный вариант.

Предполагаемый вариант построения ЦПС:

- отсутствующий ЦПС как совокупность слов-прототипов берется из проблемно-ориентированного САПР; данный случай предполагает автоматическое создание грамматики для ЦПС на базе набора слов.

4) Пользователь не способен построить ни ЦПС, ни ПОС.

Предполагает варианты построения ЦПС:

- отсутствующий ЦПС как совокупность слов-прототипов берется из проблемно-ориентированного САПР; данный случай предполагает автоматическое создание грамматики для ЦПС на базе набора слов.

Предполагаемый вариант построения ПОС:

- отсутствующий ПОС строится автоматически на базе грамматики, задающей ЦПС.

В соответствии с данной классификацией рассмотрим порядок выполнения указанных операций синтеза внутренней структуры поблоков, а так же состав параметров синтеза для различных режимов построения модуля знаний.

#### **3.4.1. Режим неявных продукций**

Режим отличается следующими чертами [10]:

- И-ИЛИ-деревья, имеющиеся на различных этапах создания модели структуры, стоятся автоматически, путем обучения;

- продукций, связывающие различные ИЛИ-синтермы как в одном И-ИЛИ-дереве, так и в двух произвольных деревьях, стоятся автоматически, используя атрибуты, имеющие место в альтернативах ИЛИ-узлов.

Синтез внутренней границы блока  $R^k_{t+1}$  по комплексу идентификаторов внешней среды блока  $W S^k_{t+1}$  выполняется следующим путем:

- выбор из множества оставшихся альтернативных вариантов структуры внутренней границы одного варианта, исходя из особенностей его структуры по соответствующему И-ИЛИ-дереву.

При этом рассматриваются:

- идентификатор всей внутренней структуры блока, построенный с учетом внешней границы блока;

- идентификатор конкретного варианта внутренней границы.

При этом идентификатор конкретного варианта внутренней границы фактически отражается через набор идентификаторов всей структуры блока в целом, заданных как атрибуты ИЛИ-альтернатив.

Вывод осуществляется на основе ряда неявных продукций, автоматически формируемых в ходе диалога синтеза структур на основе атрибутов, принадлежащих отдельным ИЛИ-альтернативам.

Множество данных, входящих во внутреннюю границу подблока, это перечисление своих и чужих свойств, включая:

- его собственных подблоков (выходы), имеющих выходные связи с блоками, относящимися к внешней среде по отношению к данному блоку;
- его собственных подблоков (инкапсулированные по выходу данные), не имеющих выходных связей к блокам, относящимся к внешней среде по отношению к данному блоку;
- чужих по отношению к данной внутренней среде блоков, т.е. относящихся к внешней среде блока и являющихся входными по отношению к данному блоку (входы), а точнее – входными по отношению к его подблокам, как к выходным, так и к инкапсулированным по выходам.

Множество параметров модуля знаний, используемых при синтезе внутренней среды  $S^k$  данного подблока, составляет:

- множество данных, входящих во внутреннюю границу подблока, что является основой для синтеза требуемой внутренней среды;
- множество ИЛИ-узлов И-ИЛИ-дерева множества связей.

Синтез  $S^k$  имеет такие этапы:

- сужение множества альтернатив для ИЛИ-узлов И-ИЛИ-дерева множества связей, исходя из особенностей состава внутренней границы подблока  $R^k$ ;
- свободный диалоговый синтез внутренней среды  $S^k$  по ИЛИ-узлам предварительно суженного И-ИЛИ-дерева множества связей.

Этап сужения множества альтернатив для ИЛИ-узлов И-ИЛИ-дерева множества связей выполняется автоматически, в ходе предыдущего этапа синтеза варианта внутренней границы за счет использования неявных продукций.

Этап свободного синтеза внутренней среды  $S^k$  по ИЛИ-узлам И-ИЛИ-дерева выполняется в диалоге, способом, аналогичным выбору варианта требуемой внутренней границы.

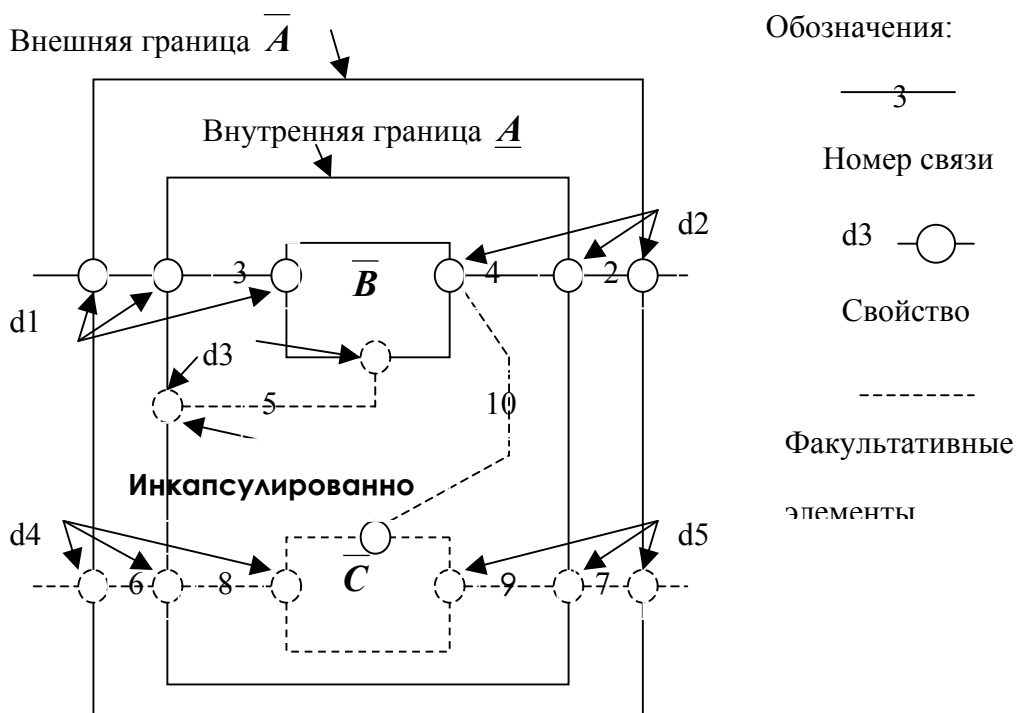


Рисунок 2 - Обобщенная схема типа А

Детально данный режим описан в [10]. Рассмотрим пример. Пусть имеется некоторое множество прототипов, входящих в тип блоков А:  $A=(P_1 \vee P_2 \vee P_3)$ . На рис. 2 показана обобщенная схема типа А, построенная в рамках предлагаемой концептуальной модели. Результат выполнения теоретико-множественных операций над совокупностями "обобщенных" связей (далее - просто связей), образующими данные прототипы, составляет ряд подмножеств @j, определяющих некоторые части внутренней среды прототипов. При этом:

$$P_1=@1\&@2; P_2=@1\&@2\&@3; P_3=@2\&@3;$$

$$@1=5; @2=1\&2\&3\&4; @3=6\&7\&8\&9\&10.$$

Тут 1,2,... - сквозные номера оригинальных связей, определенные в пределах всего типа блоков. Преобразуем множество прототипов А к форме И-ИЛИ-дерева:

$$A=(P_1 \vee P_2 \vee P_3) = @2 \& N1; N1=@1 \vee @3 \vee N2; N2=@1\&@3.$$

На рис. 3 изображено полученное И/ИЛИ дерево. В скобках показаны номера прототипов, входящих в данную вершину, числами заданы номера связей, стрелками показан порядок декомпозиции узлов.

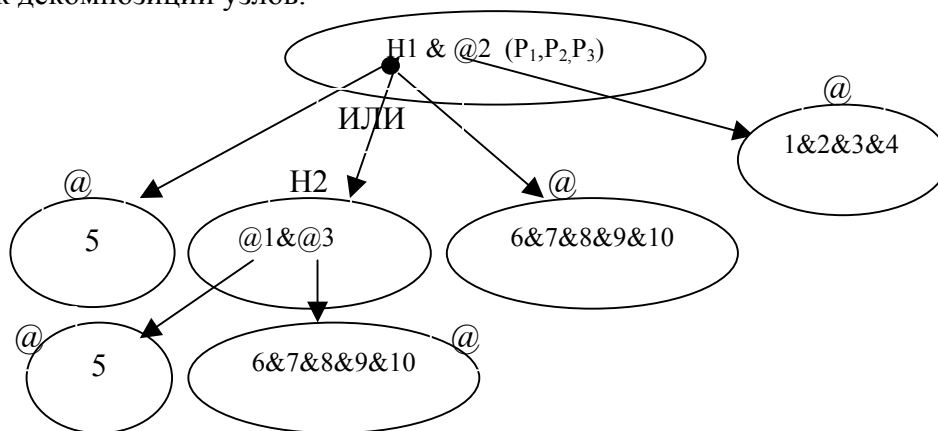


Рисунок 3 - Форма И-ИЛИ-дерева

Номера прототипов в вершинах неявно задают продукционные зависимости для ИЛИ-узлов, используемые при выводе с целью синтеза (выбора) необходимых прототипов в САПР. Семантика зависимостей такова: "если в данном ИЛИ-синтерме (узле) удалить вариант (признак), включающий прототип К, то прототип К должен быть удален во всех прочих узлах". С учетом названных зависимостей А может быть определен как идентификатор модуля знаний о структуре внутренней среды объектов типа А.

### 3.4.2. Режим явных продукций

Отличается от режима неявных продукций следующими чертами :

- задание напрямую И-ИЛИ-дерева, имеющиеся на различных этапах создания модели структуры;
- задание напрямую продукций, связывающих различные ИЛИ-синтермы как в одном И-ИЛИ-дереве, так и в двух произвольных.

Т.о., явные продукции напрямую реализуют перечисленные выше зависимости и этапы синтеза внутренней структуры блока на основе И-ИЛИ-деревьев, построенных прямым вводом.

Детально данный режим описан в [11].

### 3.4.3. Режим пакетных продукций

Отличается от режима неявных продукций тем, что [11]:

- все зависимости, задаваемые неявными продукциями через атрибуты ИЛИ-альтернатив задаются явно, в пакетных продукциях;



- отсутствуют отдельно построенные И-ИЛИ-деревья, имеющие место на различных этапах создания модели структуры; они неявно отражены в посылках и выводах неявных продукций.

Т.о., пакетные продукции напрямую реализуют перечисленные выше зависимости и этапы синтеза внутренней структуры блока на основе И-ИЛИ-деревьев, косвенно задаваемых в продукциях.

Детально данный режим описан в [12]. Рассмотрим пример.

Для передачи системе информации о порядке выбора продукций описываются правила вывода. Грамматика языка описания правил вывода имеет такие особенности. Каждое правило вывода имеет свое уникальное имя. В правой части правила описывается выражение, аргументами которого являются имена продукционных правил и имена самих правил. В правой части могут быть использованы скобки и отношения "и" и "или". Пример формата правила приведен на рис. 4.

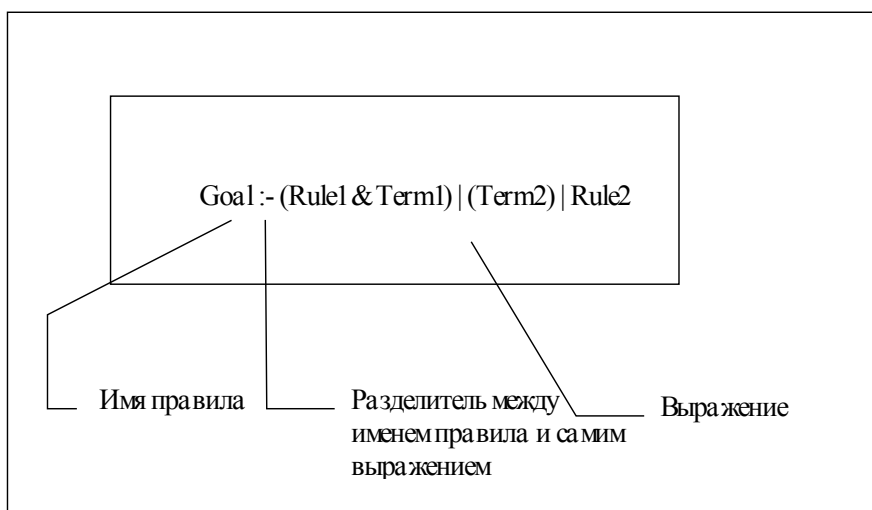


Рисунок 4 - Пример формата правила

В выражении допускается использовать все имена продукций, описанных в проекте. Правилем цели является правило, описанное в начале. Правила перед обработкой переводятся в полиз, а далее по полизу строится "и-или" дерево вывода. В вершинах дерева находятся имена продукционных правил.

Формат продукции, принятый в системе следующий:

Имя Продукции: "Условная часть" Тип Продукции "ТО-часть".

Имя продукции - это строка символов, необходимая для управления порядком обработки продукций. "Условная" и "ТО" часть продукции имеют идентичный формат. Как "условная" так и "ТО" часть состоят из элементов. Элементы представляют собой нисходящий уровень описания детализации структуры проектируемого объекта по формату: {Название\_библиотеки.Тип.Массив.Номер.Свойства.}. В процессе обработки продукции для "условной" части проверяется наличие элементов в БД. Если "условная" часть истинна, то элементы "ТО"-части создаются в БД.

Например, приведем фрагмент описания прямого участка трубопровода:

Трубы.Прямые.Трасса1.0.материал=чугун.

Этот фрагмент может быть как "условной" так и "ТО"-частью в продукции. Здесь: трубы - название библиотеки, прямые - тип, трасса1 - массив, 0 - специальный идентификатор номера элемента массива, подразумевающий весь массив, свойство - материал=чугун.

Продукции могут быть определенными и недоопределенными. Если для одного из элементов продукции не существует ни одного известного решения, то на место имени элемента ставится зарезервированное слово New. В этом случае значение элемента будет определяться в диалоге с пользователем в процессе отработки продукции. Использование зарезервированного слова NIL предполагает выбор пользователем в диалоге любого значения из числа возможных для данного элемента. Например:

Трубы.NIL.Трасса1.0.свойство=чугун.

В ходе отработки продукции вместо NIL будет браться значение, которое выберет пользователь: Прямые или Расширяющиеся или Изогнутые ...

На рис. 5. представлен фрагмент диалога отработки недоопределенных изделий посредством обращения к динамической БД.

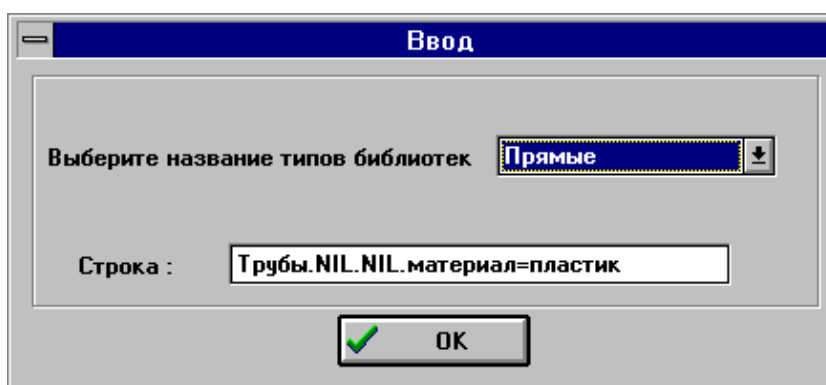


Рисунок 5 - Фрагмент диалога отработки изделий

Между "условной" и "ТО" частями стоит символ, определяющий тип продукции. Возможные варианты:

1. "->" - порождает элемент со своими свойствами, например:

d : Трубы.NIL.NIL.NIL.материал=чугун ->

Трубы.Колено.New.NIL.Материал=Керамика;

Продукцию следует понимать так : если существуют трубы с материалом - чугун, то создать трубу типа колено с материалом керамика.

2. "=>" - порождает связи в проектируемой модели, например:

f : Насос.NIL.NIL.0.Мощность=200Вт =>

Трубы.Прямые.Трасса2.NIL.Соединение=Сварка;

Продукцию следует понимать так: если существует насос, мощностью=200Вт, то существует элемент “Трубы.Прямые.Трасса2.соединение=сварка” и структурная связь между названными элементами.

3. Безусловная продукция. Она выполняется всегда и в ее теле отсутствует "условная" часть. Пример:

NIL.NIL.NIL.NIL.стоимость=100\$

и т.д.

В продукциях на месте значения элемента [свойства] (после знака "=" ) могут стоять: произвольные строки; действительные числа; переменные (их значения вычисляются в процессе разбора); математические функции или выражения; скобки. Переменные, математические функции и выражения должны стоять внутри символов "<" ">". Переменные могут описываться отдельно от изделий между символами # # и им может присваиваться требуемое значение с помощью знака ":= ". Переменные могут описываться и

в самих продукциях, перед значением элемента [свойства], после знака "=". Комментарии записываются между "(" и ")"".

Каждый элемент может иметь произвольное количество значений. Например, значение "типа" для данного примера с трубопроводами может быть - "конусообразные", "прямые" и т.д. В элементе продукции [свойства] так же может быть множество вариантов значений, перечисленных через запятую (после знака "=").

Пример :

# i:=1.2345 #

Прод1 : Труба.Прямая.Трасса1.3.диаметр=100мм,130мм,150мм ->

Насос.Водяной.NIL.1.мощность=< 200\*sin(sqrt(i)) >;

Здесь: между символами # # описывается переменная i и ей присваивается начальное значение 1.2345; Прод1 - имя продукции.

Понимать эту продукцию следует так:

ЕСЛИ существует Труба, Прямая, В Трассе1, под номером 3, диаметром=100мм или 130мм или 150мм;

ТО создать Насос, Водяной, недоопределенного массива, номер 1, мощностью = 200\*sin(sqrt(i)).

Наличие " Трубы Прямой Трасса1 " со свойством "диаметр=100мм" проверяется в БД. И если она имеет место, то это значит, что условная часть продукции истинна и будет срабатывать "ТО" часть. Так как символ '->' указывает на то, что продукция порождает элемент без связи, то порождается "Насос Водяной" со свойством "мощность=...".

Значение свойства "мощность" вычисляется автоматически по формуле  $200 \cdot \sin(\sqrt{i})$ . Значение свойства может иметь и текстовый вид, например: "мощность=большая, маленькая". В данном примере через запятую перечислены все варианты, которые может приобретать это свойство.

Наличие вариантов по ИЛИ в продукциях определяет их как макропродукции. Порядок отработки макропродукций предполагает использование механизмов нумерации и взаимного отображения вариантов посылок и "ТО"-частей продукции друг в друга. Нумерация вариантов осуществляется по стандартному методу обхода дерева вариантов «сверху-вниз» и «слева-направо».

Метод взаимного отображения вариантов предполагает взаимно-однозначное соответствие одинаковых номеров в случае одинакового числа вариантов и полное покрытие большего числа вариантов меньшим путем повторения, в случае их разного количества в посылке и действии.

В случае, если при отработке продукции обрабатывает более чем одно условие посылок, то формируется дерево вывода, приводящее к порождению ряда решений. Дерево вывода - представляет собой динамическую структуру данных. Особенности дерева вывода таковы: дерево вывода - симметричное, бинарное (каждая вершина дерева имеет одинаковое число исходящих дуг, равное двум), несбалансированное (в нем могут быть не заполненные уровни). Для построения дерева используется механизм полза.

### **Выводы.**

В предлагаемой работе:

1) Приведены принципы:

- разделения единого модуля описания модели объекта на различные подмодули (отдельные И-ИЛИ-деревья) описания: внешней границы, внутренне границы, внутренней среды (функция как частный случай внутренней среды), отдельных групп связей, групп свойств и т.д.;

- введения сложной идентификации элементов модели, а не только прототипов в целом.

2) Предложена формальная постановка задачи синтеза структур на основе данной расширенной семиотической модели;

3) Предложены методы решения поставленной задачи синтеза, адаптируемые к уровню квалификации эксперта в данной предметной области.

Предлагаемые методы и средства обеспечили создания инструментальных средств построения интеллектуальных САПР для этапа структурного синтеза сложных объектов. Функционируя по принципу “отдельная предметная область - отдельная библиотека”, предлагаемая система способна обеспечить создание ряда специализированных интеллектуальных САПР, обеспечивающих синтез структур. Для обеспечения накопления знаний системой в настоящее время выполнено построение следующего ряда интерфейсов:

- с системой автоматизации проектирования парогазовых установок для теплоэлектростанций - СПРУТ и, через ее посредство, с САПР AUTOCAD;
- с САПР проектирования средств вычислительной техники Or-CAD с языком VHDL;
- с системой автоматизации проектирования экономических проектов Project-Expert;
- с табличным процессором EXCEL;
- с рядом других проблемно-ориентированных САПР.

### Литература

1. Техника, которую мы потеряем. Комсомольская правда в Украине. 23.12.2009. Режим доступа <http://kr.ua/daily/231109/203259/>.
2. Григорьев А.В. Представление недоопределенности знаний в инструментальной оболочке для построения САПР. Искусственный интеллект. N 1, 1999. — С. 96-106.
3. Григорьев А.В. Система уровней модели предметной области для интеллектуальных САПР, Москва: Физико-математическая литература, КИИ'2000, 2000, С. 633-642.
4. Григорьев А.В. Семиотическая модель базы знаний САПР. Научные труды Донецкого государственного технического университета Серия: Проблемы моделирования и автоматизации проектирования динамических систем, выпуск 10: Донецк, ДонГТУ, 1999. — С. 30-37.
5. Солодовников В.В., Тумаркин В.И. Теория сложности и проектирование систем управления. - М.: Наука. 1990. - 186 с.
6. Поспелов Д.А. Ситуационное управление: теория и практика. М.: Наука, 1986. - 288 с.
7. Григорьев А.В. Содержание некоторых категорий абстракций в теории построения интеллектуальных САПР/ Наукові праці Донецького національного технічного університету / Редкол.: Башков Є.О. та інші. Серія: "Обчислювальна техніка та автоматизація": Випуск 64.: Донецьк: Видавництво ДонНТУ, 2003 - С. 166-178.
8. А.В. Григорьев. Организация пространства поиска решений в специализированной оболочке для создания интеллектуальных САПР. Вісник ТРТУ-ДонГТУ. Матеріали 2-го міжд. н.-т. семінара "Практика і перспективи інституціонального партнерства". Донецк, ДонГТУ, 2001, N1, С.57-68.
9. Григорьев А.В. Пути создания интеллектуальных САПР при различных уровнях квалификации экспертов /Научно-теоретический журнал «Искусственный интеллект», №3, 2005. – Донецк: ИПИИ МОН и НАН Украины «Наука и образование», 2005. – с. 758–763.
10. Григорьев А.В. Специфика выполнения теоретико-множественных операций над контекстно-свободными грамматиками в условиях различных форм дополнительных семантических правил в семиотической модели интеллектуальных САПР / Научные труды Донецкого национального технического университета. Серия «Проблемы моделирования и автоматизации проектирования динамических систем» (МАП – 2006). Выпуск 5 (116). – Донецк: ДонНТУ, 2006. – С. 91-104.

11. Григорьев А.В. Классификация типов продукций в интеллектуальных САПР / Наукові праці національного технічного університета. Серія «Обчислювальна техніка та автоматизація». Випуск 88. –: Донецк: ДонНТУ, 2005. – с. 99-105.
12. Григорьев А.В., Базалей А.О. Специализированная оболочка для синтеза интеллектуальных САПР и АСНИ. В кн. Информатика, кибернетика и вычислительная техника (ИКВТ-97). Сборник трудов ДонГТУ, Випуск 1. Донецк: ДонГТУ, 1997. С. 225-228.

#### **Abstract**

**Grigoriev A.V. Method of the decision of a problem(task) of structural synthesis in intellectual САПР, the structures constructed on the basis of semiotics model.** In work are described formal statement of a task of synthesis of structures on the basis of semiotics model of structures, and as methods of the decision of the given task. The set of methods of the decision of a task provides adaptation on a skill level of the expert in the given subject domain. Each mode assumes a corresponding way of construction of base of knowledge and the method of the organization of a conclusion.

**Key words:** semiotics model of structures, formal statement of a task, the module of knowledge, OR-END-tree, production, adaptation on qualification of the expert, modes of a conclusion.

#### **Анотація**

**Григорьев А.В. Методи вирішення завдання структурного синтезу в інтелектуальних САПР, що побудований на базі семіотичної моделі структур.** В роботі описується формальна постановка завдання синтезу структур на основі семіотичної моделі структур, а також методи рішення даної задачі. Набор методів рішення задачі забезпечує адаптацію на рівень кваліфікації експерту у даній предметної області. Кожен режим має відповідний спосіб побудови бази знань і свій метод організації виводу.

**Ключові слова:** семіотична модель структур, формальна постановка задачі, модуль знань, I-АБО-дерево, продукції, адаптація на кваліфікацію експерту, режими виводу.

Здано в редакцію:  
16.04.2010 р.

Рекомендовано до друку:  
д.т.н., проф. Башков Є.О.